NNRO »SPORTSKA TRIBINA«

mr®

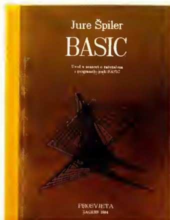
Revija za MALA RAČUNALA

Broj 1 God. 1
Cijena 250 din



- SPECTRUM
- COMMODORE – 64
- ZX 81
- GALAKSIJA
- BBC
- SPERRY PC

Biblioteka POPULARNA INFORMATIKA



JURE SPILER:

BASIC

Knjiga je namijenjena svima koji se žele upoznati s najpopularnijim programskim jezikom za kućna računala, ali i onima koji već dobro poznaju BASIC.

CIJENA 950.– DIN

Komplet **SVIJET KUĆNIH RAČUNALA**

1. SVE O KUĆNIM RAČUNALIMA
2. PRVI KORACI U BASICU
3. IGRE, GRAFIKA I ZVUKOVI
4. UČENJE UZ RAČUNALO

CIJENA KOMPLETA
U PRETPLATI 4.000,– Din.



Prva knjiga izlazi tokom studenog, dok će se ostali naslovi pojaviti početkom 1985.

Komplet je vrlo pogodan za uvođenje u sve tajne danas toliko popularnih mikrokompjutera. Svaka je knjiga pažljivo napisana i ilustrirana, i zbog toga vrlo čitka, informativna i puna praktičnih savjeta.

Narudžbenica za navedene knjige štampana je na 64 (zadnjoj) strani.

Uvodna riječ urednika

Kada je prije nekoliko godina započeo nagli razvitak mikroračunala i ostale popratne opreme, vjerojatno niti sami tvorci nisu predviđali naglu eksploziju koje smo danas sudionici. Ova najnovija, današnja, tehnološka revolucija juri još dosad neviđenom brzinom i nitko razuman ne želi se upuštati u dugoročnije prognoze jer ono što danas napišemo može biti već za nekoliko mjeseci zastarjelo i izgledati diletantski, nestručno i smiješno.

Danas se o mikroračunalima govori i piše posvuda. U naše domove iznenada se uvukla groznica koja već pola desetljeća potresa zapadni svijet. Stigla je na mala vrata, ali nezaustavljivo. Nema dana a da se nešto o računalima ne objavi u dnevnom tisku, na radiju ili televiziji. Već postoje časopisi specijalizirani za problematiku miniračunala, a u mnogim drugim časopisima postoje stalne ili povremene rubrike o toj tematici.

Pa koji su onda razlozi za izdavanje još jednog časopisa? Pri definiranju naslovne stranice htjeli smo publikaciju nazvati »revijom za praktičnu primjenu mikroračunala«.

Nedavno je objavljena crtica iz uваženog engleskog časopisa u kojoj se navodi kako većina ljudi koji kupe kućno računalo odlažu vrlo brzo novog ljubimca na najvišu policu. Naravno, radi se o odraslim zaposlenim ljudima koji nemaju vremena baviti se izučavanjem i programiranjem, što ponekad i nije jednostavno. I kad čar novosti mine, a igre postanu monotone (ima ljudi koji se uopće ne vole igrati) nekadašnju novost prekriva zaborav. Ali ima i onih koji uvučeni u čudesni svijet bajke i kombinatorike bivaju za sva vremena zarobljeni u njihovim raznovrsnim labirintima.

Jedan dio, posebno mladih i nadarenih, ne želi se zaustaviti samo na obodima nastale eksplozije. Oni žele proniknuti sve dublje u samu suštinu cjelokupne revolucije koja mijenja svijet. Nasuprot njima, postoji generacija trideset i četrdesetogodišnjaka koji shvaćaju da su »kompjutori nešto važno« i da o njima nešto treba znati jer će ih inače vrijeme jednostavno pregaziti. Bez obzira na nedostatke u procesu obrazovanja i nedovoljnoj informiranosti (uz ostalo i nedovoljno kvalitetnoj), mlade generacije prožete su znanjima na području informatike.

Naša je želja da pripomognemo, u današnje vrijeme toliko razvikanom, ovladavanju tehničkom kulturom u korištenju računala. I ono, kao i sva ostala tehnička (i ne samo tehnička) dostignuća, može služiti dvjema svrhama: što snažnijem očovječenju čovjeka ili dehumanizaciji. Sve ovisi o pristupu korisnika. Na čovjeku je da odluči hoće li biti rob ili gospodar vlastitih izuma. Mi smo za ovo posljednje.

Željeli bismo uz vašu suradnju dragi čitaoci, pronaći što raznovrsniju primjenu ovog rezultata rada čovjekovog uma.
Ne želimo previše teoretizirati, već bismo htjeli da svaki članak posluži za nešto praktično. Kako su područja zanimanja čitalaca različita, i raspon članaka je širok i šarolik. Pokušali smo pronaći za svakog ponešto. Nekom će poneki članak biti tek ponavljanje nečeg već mnogo puta prorađenog, nekom će poneki članak biti prestručan i nerazumljiv. No, bez obzira na tematiku članka nastojali smo problematiku razmotriti na što razumljiviji način.

Jesmo li uspjeli? U čemu smo griješili? Pomozite nam i javite se. Bez vašeg odgovora lutat ćemo u tami. Pomozite nam da bismo i mi vama mogli pomoći. Javite se, zovite, dođite.

Uspjet ćemo tek onda ako SVAKI čitalac pronađe bar jedan članak koji mu je pomogao u savladavanju bar jednog od njegovih problema. Želimo vam pomoći da vaše računalo ne ostane jedna od tehničkih naprava koja je »plesala samo jedno ljeto« već da bude prijatelj koji vam olakšava i uljepšava život.

Urednik

SPORTSKA TRIBINA

U ovom broju

Uvodna riječ urednika 1
U ovom broju 2

Prikazi računala

Sperry PC 3
Atari 600 XL i 800 XL 6
Oric Atmos 7
Merit 8

Programska podrška (Software)
Oprema (Hardware)

Pismo iz Silicijske doline 9
A šta on može? 10
Računalo upravlja jedrilicom 11
Prvih 10 godina 15
»Prosvjetin« računalski happening 18
Najnovije: ZX Spectrum 19
Što treba znati pri kupovini »mikri-ća« 20
Evolucija mikroprocesora 21
Razvrstavanje (sortiranje) na malim računalima 24
Spectrumove tajne: Ekran 28
Palice za igru i njihovi priključci 30
Savjeti za brzi Basic 34
Računala mogu vidjeti 35
Elektronični glazbeni instrumenti 36
Programski tumač ili programski prevodilac 38
Najznačajniji proizvođači BASIC prevodioca 39

Biblioteka

Programska biblioteka za početnike--Spectrum 40
Editor za definiranje funkcijskih tipki – BBC 46
Commodore-64, Vic-20 Commodore 49
Program za brzo učitavanje i zapis na kazetu ZX-81 (16K) 52
Isprobajte! ZX-81 54
Igra koja dolazi (Kokotoni Wilf) 55
Programi u jednoj liniji-Spectrum 56

Uradi sam

Rubna spojnica (Edge connector) za računala Galaksija 58
Fizičko brisanje memorije i spreča-vanje pregrijavanja ZX Spectruma 60
Kazetni adapter-Vic 20 61

Igre

Top-10 62
Ronilac Scuba
Smetlar (Trashman) 63

SPERRY PC

Ovim napisom želimo vam predstaviti novog člana porodice Sperryjevih računala, kojim se ta firma uvrstila u grupu proizvođača osobnih računala – konkurenata IBM-ovom PC-u.

U proizvodnom programu Sperryja unazad desetak godina postoje inteligentni terminali od kojih je najpoznatiji UTS 400 (*Universal Terminal System*). Kontrolna jedinica UTS-a 400 je Intelov 8-bitni mikroprocesor 8080, a od periferijskih jedinica opskrbljen je disketnim jedinicama i pisaćem. Međutim, ti terminali su zavisni od veze s većim računalom (1100 serija Sperryjevih računala) jer je rad na njima podijeljen u 2 faze: prevođenje simboličkog jezika (assembler, COBOL, . . .), objedinjavanje, (povezivanje grupe programa u jedan koji se izvršava), punjenje takvog programa u memoriju terminala što se sve izvodi na velikom računalu (*on-line*) i druga faza – izvođenje programa što je nezavisno od veze s velikim računalom (off-line).

Takav tip terminala vremenom je razvijen do modela UTS 30 i UTS 40. Budući da su prije dvije godine osobna računala postala pravi hit na tržištu, svi su se veliki proizvođači računala, pa tako i Sperry, nastojali uključiti u proizvodnju. Stoga je inteligentnim terminalima Sperry dodao mogućnost samostalnog rada, odnosno upravljački sistem (CP/M, CP/M Plus), a samim tim i širok repertoar programske podrške kao npr. CBASIC, CIS COBOL, Pascal, PL/1-80, kao i mogućnost izvođenja raznih programa za podršku bazama podataka, grafici itd.

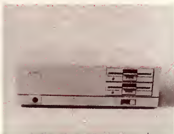
No, u međuvremenu su se pojavila osobna računala razvijena na bazi 16-bitnih mikroprocesora (IBM PC, Olivetti PC, Hp 150 itd.), s mnogo više memorije (memorija je postajala sve jeftinija i sve manje ograničavajući faktor u cijeni računala), a time i većih mogućnosti. Trebalo se uključiti i u tu konkurenciju. Tako se početkom ove godine na američkom tržištu pojavio SPERRY PC, po cijeni i tehničkim karakteristikama direktni konkurent najpoznatijem osobnom računalu te klase IBM PC-u.

Na ovogodišnjem sajmu »Interbiro-informatika« održanom u Zagrebu od 8. do 12. 10. bio je na štandu »Infosistema« iz Zagreba, koji je generalni zastupnik Sperryja za Jugoslaviju, izložen jedan od modela Sperry PC-a i pobudio je veliko zanimanje kako laičke publike koja se oduševljavala grafičkim mogućnostima monitora visoke raščlanjenosti, tako i publike koja je pokazala interes za nabavku tog računala.

U tehničkom opisu Sperry PC-a naći ćemo dosta zajedničkih svojstava s IBM PC-om i ostalim osobnim računalima istog tipa (16-bitni mikro-procesor), no to i jest ono što se zahtijeva jer bitno je napraviti osobno računalo tako da može izvoditi programe bilo kojeg drugog. Drugim riječima treba biti IBM primjenjiv (*compatible*). U tom smislu je i nastala izreka: »Danas čak i IBM treba nastojati da bude IBM primjenjiv«.

Sperry PC se kao i ostala osobna računala sastoji od kontrolne jedinice, monitora (video ekrana), tastature, jedinice vanjske memorije i štampača. Proizvodi se u 7 modela koji se međusobno razlikuju po karakteristikama monitora i tipu vanjske memorije (diskete, Winchester disk ili kombinacija).

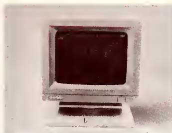




Slika 1



Slika 2



Slika 3



Slika 4

Mikroprocesor kontrolne jedinice je 16-bitni Intelov 8088. Opskrbljen je malom tehničkom dosjetkom. Naime, moguće je birati brzinu rada 4,77 MHz ili 7,16 MHz (što ubrzava rad za 50%).

Korisnička memorija (*RAM*) ima 128 K i vrijeme ciklusa od 270 ns, vrijeme pristupa 150 ns i može se proširiti do 640 K.

Takvo proširenje je predviđeno praznim mjestima u kućištu kontrolne jedinice, tako da se memorija može povećati jednostavnim umetanjem pločice (feature card) u za to predviđeno mjesto.

Stalna memorija (*ROM – Read Only Memory*) sadrži programe za testiranje rada dijelova računala i BIOS (Basic Input-Output System) – skup programa za izvođenje ulazno-izlaznih radnji s tastaturom, ekranom, vanjskom memorijom i pisačem.

Računalo ima sat koji se napaja baterijom i koji održava datum i vrijeme dana, te zvučnik za zvučne efekte i muzičku primjenu. Što se tiče komunikacija, tu je standardni međusklop RS232C – uređaj koji omogućuje asinhronu vezu preko komunikacijskog kanala brzinom od 50 do 9600 bitova u sekundi i u formatu od 5, 6, 7 ili 8 bitova. Uz kontrolnu jedinicu u istom kućištu se nalaze i jedinice vanjske memorije, što je proizvođač zajedno nazvao sistemskom jedinicom (*system unit*). Podržavaju se dva tipa vanjske memorije: jedinica dvostranih disketa od 5,25 inča dvostruke gustoće i fiksni disk od 10 megabajta. Vrsta i broj jedinica vanjske memorije zavise o tipu računala (slika 1). Diskete su formatizirane tako da mogu primiti 360 KB ili 164 tiskane kartice. Za one kojima je potrebna veća trenutno dostupna memorija, rješenje je fiksni disk opskrbljen vlastitim mikroprocesorom koji vrši prijenos, pronalaženje i otklanjanje grešaka.

Svaki od modela ima ergonomički konstruiranu tastaturu od 84 tipke, takozvanu potpuno funkcionalnu (*full-function*) tastaturu što znači da se svaka tipka može programirati. Slična je onoj kod IBM-ovih modela (*naravno QWERTY*) s tim da je opremljena s dodatnih 10 funkcijskih tipki, odvojenim numeričkim tipkama i signalnim svjetlima nad LOCK tipkama.

Tastatura je odvojena od sistemske jedinice i priključuje se dva metra dugim kablom, što omogućuje smještaj po korisnikovoj želji. Monitor ili video ekran proizvodi se u tri varijante: crno-bijeli, monitor srednje raščlanjenosti i visoke raščlanjenosti. Crno-bijeli ili bolje rečeno crno-zeleni monitor ima ekran dijagonale 12 inča (30 cm). Znak se sastoji od 7×9 točaka unutar bloka od 9×14 točaka. Sposoban je prikazati 25 linija po maksimalno 80 znakova, te posjeduje kontrole svjetloće i kontrasta. Pod kontrolom programa moguće je mijenjati boju pozadine i znaka unutar bloka, aktivirati blinkanje znaka te potcrtavati. Svaki od monitora posjeduje vlastiti mikroprocesor i memoriju ekrana, te mogućnost prikaza 256 znakova koji uključuju 96 ASCII grafičkih znakova te simbole za obradu teksta i poslovne izvještaje. Tako crno-bijeli monitor ima 4KB memorije i posebni međusklop za pisač spojiv na Centronics međusklop, što daje mogućnost priključenja velikog broja pisača raznih proizvođača. Monitor srednje raščlanjenosti može prikazati 256 znakova u 16 boja na pozadini u jednoj od 8 boja. Što se tiče grafike, ekran se sastoji

od 320×200 točaka, čija se boja i jačina kontroliraju nezavisno. U visokoj raščlanjenosti (640×200 točaka) monitor postaje crno-bijeli.

Monitor visoke raščlanjenosti, prikazan na Interbirou pruža izvanredne mogućnosti grafičkih prikaza. Postoji više kombinacija raščlanjenosti i broja boja, tako npr. u visokoj raščlanjenosti 640×400 točaka moguće je prikazati 16 boja iz skupa od 256 boja, a u smanjenoj raščlanjenosti 320×400 odjedanput svih 256 boja. To omogućuje memorija ekrana od čak 192 KB. Ovaj monitor može prikazati također 256 znakova i to u 25 linija, te 80 ili 40 stupaca. Veličina memorije omogućuje miješanje grafičkog prikaza i teksta na ekranu.

Uz svaki model mogu se priključiti dva tipa pisača: model 5 i model 31.

Model 5 štampač je točkasti (*dot-matrix*) u stvari Epson RX-80) dvostrani štampač brzine 100 znakova u sekundi. Znak je sačinjen od 9×9 točaka, a u jednoj liniji može odštampati 40 do 137 znakova iz skupa od 235 znakova. Ovaj štampač se priključuje ili preko crno-bijelog monitora ili preko međusklopne pločice za štampač ako je monitor u boji.

Drugi tip, Model 31 je pisač koji odjednom otiskuje cijeli znak (DAISY-WHEEL). Piše brzinom od 55 znakova u sekundi, također dvostrani. Štampa 132 znaka u liniji kod gustoće 10 znakova po inču i to iz skupa od 96 znakova. Za priključivanje vrijedi isto kao i kod modela 5.

Preostaje još da navedemo neke od mogućnosti proširenja, koje se postiže dodavanjem pločica u sistemsku jedinicu. To su, kao što smo već naveli, proširenje korisničke memorije do 640 K bajtova, dodavanje pločice kojim se postiže da Sperry PC oponaša SPERRY UNISCOPE terminal ili IBM 3270 bisinhroni ili SNA terminal, te dodavanje pločice na kojoj je poznati Centronics međusklop za spajanje pisača. Nakon tehničkog opisa treba nešto reći i o programskoj podršci računala. Danas u svijetu postoje mnoga poduzeća koja se bave pisanjem programske podrške za određeni tip mikroprocesora kao npr. Microsoft, Lotus, Digital Research itd. Tako je i za Intelov 8088 mikroprocesor napisana golema količina raznih paketa i aplikacija počevši od upravljačkih sistema, pa do raznih igara. Zahvaljujući pojmu IBM primjenjivosti svi se ti programi mogu izvoditi i na Sperry PC-u. U nastavku ćemo kratko opisati osnovne programe i najpoznatije aplikacije Sperry PC-a.

MODEL	Standardna memorija	Diskete	Disk	Monitor	Pisač
10	128 K	1 320K	–	C/B	Model 5 ili 31
20	128 K	2 320K	–	C/B	Model 5 ili 31
25	128 K	2 320K	–	Srednje rašč.	–
30	128 K	2 320K	–	Visoke	–
40	128 K	1 360K	1 1OMB	C/B	Model 5 ili 31
45	128 K	1 360K	1 1OMB	Srednje rašč.	–
50	128 K	1 360K	1 1OMB	Visoke rašč.	–

Upravljački sistem je Microsoftov MS DOS, vrlo jednostavan po strukturi i upotrebi, a osnovni jezik je trenutno BASIC za DOS 2.11, jezik s mnogo dodatnih mogućnosti, pogotovo u dijelu grafike i rukovanja datotekama. Uz BASIC koji se isporučuje s računalom, može se nabaviti i COBOL, FORTRAN, Pascal, PL/1 i Turbo Pascal, programe prevodioce. Što se tiče obrade teksta, u samom upravljačkom sistemu postoji program EDLIN uglavnom namijenjen pisanju simboličkih programa. Ako je namjena PC-a isključivo obrada teksta, može se nabaviti jedan od mnogih editora kao na primjer Wordstar, Word, itd. Od baza podataka tu je Dbase II i III, Friday itd., a od komercijalnih aplikacija Lotusov 1-2-3, Multiplan, Visicalc itd. Ne treba više nabrajati jer se na SPERRY PC-u može izvoditi sve što i na IBM PC-u.

Može se zaključiti da je ovo izvanredno oblikovano osobno računalo (što je možda subjektivna ocjena autora) dostojan suparnik svim ostalim osobnim računalima ove generacije u njihovoj klasi.

Ernest Kogelman



Merit

Meritovi sistemi Merit 2000 i 3000, pojavili su se kao novost na tržištu. Zanimljivo je da mu je popularnost daleko veća u Francuskoj, Njemačkoj, Austriji, Španjolskoj i Grčkoj nago u Engleskoj.

Posebnost Merita u odnosu na ostale sisteme iste klase ogleda se u jednostavnosti primjene programske podrške. Razvoj računalske tahnike započeo je u Velikoj Britaniji i Sjedinjenim Američkim Državama, pa su sve naredbe u programima i programskim jezicima, kao i poruke, na engleskom jeziku.

Ljudi koji se profasionalno bave elektroničnom obradom podataka poznaju engleski dobro, pa im rad za tarminalom na predstavlja problem. Smanjivanjem cijena računala, ono je postalo dostupno gotovo svakom. Rad za tarminalom težak je ljudima koji ne znaju angleski jezik.

Ovaj problem riješio je MERIT. Umjesto komuniciranja na engleskom jaziku korisnik »razgovara« s računalom na materinskom jeziku. Naredbe i poruke (cjelokupni rad) odvija se jezikom korisnika. Ovakav način rada sigurno bi s odobravanjem prihvatili i mnogi jugoslavenski korisnici.

Vidanka Ribičić

TEHNIČKE OSOBINE

PROCESOR
- 8 Mz 8088 16-bitni mikroprocesor
- 128 K memorije sa vremenom
- mogućnost proširenja do 1 megabajta

VANJSKE JEDINICE
- 3 komunikaciona izlaza
- 2 disketne jedinice sa 16 megabajtom memorije
- tvrdi (Winchester) disk, 5-50 megabajta

EKRAN
- pomičan sa mogućnostima pomicanja, okretanja, nagibanja
- grafika visoke raščlanjenosti
- 256 programskih karakteristika za evropske i grafičke znakove
- 85 kolona * 25 redova
- mogućnosti pulsirajućeg prikaza znakova, obrnutog na podlozi, podvlačenja, podešavanja kontrasta

TASTATURA
- 108 sa 30 funkcijskih tipki
- tipke se mogu programirati

STANDARDNI PISAČ
- dvosmjarni sa brzinom od 100 zn/sek
- 128 znakova

PROGRAMSKA PODRŠKA

UPRAVLJAČKI SISTEM
- CP/M86 standardni upravljački sistem
- CP/M i MS DOS kao dodatak

PROGRAMSKI PAKETI
- kupci i fakturiranje
- dobavljači
- saldo konti
- financijsko poslovanje
- kontrola zaliha
- praćenje proizvodnje
- troškovi poslovanja
- osobni dohoci
- poslovno dopisivanje
- uređivanje teksta

OCJENA

Razina usavršenosti	●●●●
Mogućnosti proširenja	●●●●
Jednostavnost upotrebe	●●●●
Kvaliteta tastature	●●●●●
Programska podrška	●●●●
Kvaliteta grafika	●●●●
Dokumentacija	●●●●
Izgled	●●●●
Sveukupna ocjena	

Vidanka Ribičić



Pismo iz Silicijske doline

Nikolaj Ivančić

Pomalo je neobično pisati članak za prvi broj časopisa koji će biti štampan na udaljenosti od cca 8000 km. Kako se radi o časopisu namijenjenom jugoslavenskim vlasnicima osobnog računala, možda nije loša tema pisati upravo o tim računalima.

Moje vlastito iskustvo s osobnim računalima datira od prve pojave mikroprocesora (KIM-1, MOS Technology). Imao sam sreću da budem sudionik prvog tečaja o upotrebi mikroprocesora (1977. u Londonu), upravo u trenutku kad je moje oduševljenje velikim računalima tipa CDC Cyber, Univac 1110 bilo na vrhuncu. Unatoč neshvatljivo velikoj razlici u upotrebi (danas je teško povjerovati da je KIM, baziran na 6502 – mikroprocesoru oko kojeg je sagrađen Apple II, imao 1K memorije i heksadecimalnu tastaturu te minimalni monitor koji je omogućavao samo apsolutno programiranje), bio sam trenutačno očaran činjenicom da je KIM moj i samo moj. Ta bitna, kvalitetna razlika između mikroračunala i velikih računala (mainframe) iako danas manje izražena, odredila je sav moj razvoj i budućnost.

Imati svoje računalo koje se može upaliti kad god se zaželi, »resetirati« bez obzira na druge korisnike (jer takvih i nema), osjećaj je koji »novi« vlasnici malih računala i ne doživljavaju u punoj mjeri. Oni nemaju bolnih uspomena o beskonačnim čekanjima pred terminalom koji ne daje odaziv ili zauzetim bušilicama na kartice (»card punch«). Razvoj računalske industrije bio je dovoljno jak da prouzroči svjetski procvat posve nove industrije, industrije mikroračunala i prateće opreme. Iako pojam što je osobno računalo nije nigdje točno određen, smatram da se radi o računalu čiji je vlasnik pojedinac, a upotrebljava se u najraznolikije svrhe. Manje je jasno što je osobno računalo u Jugoslaviji. Sjećam se da je gotovo svaka republika imala svoju industriju računala (»šrafciger« industriju), vezanu na jednu od velikih svjetskih proizvođača, proizvodeći tako basnoslovno skupe uređaje iz kategorije tzv. miniračunala po cijenama iz kategorije »mainframe«. Mikroračunala su ostala na milost i nemilost malih firmi, raznih osnovnih djelatnosti i kvalifikacija, čija je osnovna motivacija bila na brzinu zaraditi novac. Privatni je uvoz bio onemogućen u potpunosti. S tugom se čitaju vijesti kako se u Mađarskoj prodaju Commodori, po cijeni koja je direktno izvedena iz odnosa dolar – forinta, bez ikakve carine, u želji da se što više rasprostrani upotreba računala. No, nije mi namjera kritizirati stanje u Jugoslaviji, već naprotiv, sa zadovoljstvom komentirati širenje upotrebe računala među »obične ljude« jer svrha tih uređaja i jest da posluže svim ljudima, u što širem krugu djelatnosti. Pri tome je od samog početka najosnovnije primiti odgovarajuću naobrazbu o njihovoj primjeni. Kako se radi o veoma složenom sustavu, treba s naobrazbom rano započeti. Nisam siguran je li konačno u Jugoslaviji definirano što je »Computer Science«, na koji fakultet pripada, i kako uopće izgleda nastavni program. Znam samo da to još prije dvije, tri godine nije bio slučaj.

Završit ću prikazom posljednje novosti na području osobnih računala koju uvodi Apple pojavom svog modela Macintosh. Osnovna je ideja i novost Mac-a da upravljački sistem predstavlja nužno zlo, pa treba biti što manje prisutan za većinu korisnika. Zakučasta terminologija (»computerise«) prelazi u manipuliranje grafičkih simbola pomicanjem mehaničkog »misa«. Kritičari takvog pristupa zasnovali su svoje nepovjerenje u novu tehniku na tvrdnji o nemogućnosti postojanja bilo kakvog »razumnog« aplikativnog softvera koji bi koristio. Čini se da Telos Software Products iz Santa Monike to uspješno opovrgava, nudeći Macintosh Data Base Management Software s nazivom Filevision. Za ilustraciju, Telos daje anatomski prikaz ljudskog oka i okružujuće muskulature, te prikaz vinskog podruma s 22 vrste boca raznih vina. Izaberemo li (pomoću misa), mrežnicu oka, Filevision će na ekranu prikazati sve informacije koje o mrežnici posjeduje. Analogno, selekcijom jedne odabrane boce saznajemo sve o dotičnom vinu. Naravno, sve je to već odavno poznato, novost je način na koji se vodi dijalog s računalom. Sudeći po uspješnosti kojom se Macintosh prodaje, novost je veoma povoljno primljena.

Mr. NIKOLAJ IVANČIĆ
dipl. ing.

Diplomirao je 1970. godine na stručnom smjeru matematičkog odjela Prirodoslovno-matematičkog fakulteta u Zagrebu, gdje je 1980. godine i magistrirao. Od 1970. do 1982. radio je u Građevinskom institutu, odnosno na Fakultetu građevinskih znanosti nakon spajanja s Građevinskim fakultetom. Bavio se primjenom programske podrške za područje građevinskog projektiranja. Od 1978. surađuje s Cromemcom, kamo prelazi 1982. godine zajedno s prof. Egonom Zakrajšekom i ing. Borisom Krtolicom, gdje su danas sva trojica među vodećim kreatorima programskih proizvoda ove cijenjene računalske tvrtke.

A ŠTO ON MOŽE?

DA BI ČOVJEK ZADOVOLJIO NEKU SVOJU POTREBU, NAJPRIJE JE TREBA IMATI!

Godine 1984. miniračunala su nahrupila u naše domove započevši istiskivati dosadašnje kućne ljubimce i mezimce kao što su televizori, HI-FI linije, pa čak i video uređaji. Proces koji se u visoko razvijenim zemljama svijeta već razbuktao, započeo je sada kod nas. Cijene ponekih računala tek su nešto više od cijena kvalitetnijih kalkulatora. Ponekad se s nevjericom i razočaranjem pristupa maloj kutijici uz uzdah: »Ah, zar je to kompjutor?«

Nakon ovakvog deprimirajućeg početka, razočarani posjetilac obično postavlja slijedeće neugodno pitanje:

»A što mogu tim kompjutorom raditi?«

Na ovakvo pitanje najčešće je nemoguće dati zadovoljavajući odgovor, posebno jer su takva pitanja postavljena na brzinu i prepad. U ovom članku pokušat ću odgovoriti na slična pitanja.

Ono što zapanjuje čak i ljude koji se profesionalno bave elektroničnom obradom podataka izuzetno je velika snaga neobično malih i začuđujuće jeftinih računala. Analiza kod nas, a i u svijetu, vjerojatno najpopularnijeg mikroračunala Spectruma ZX potvrđuje sve navedene tvrdnje:

a) Veličina memorije dostupne korisniku iznosi 48K što znači otprilike 48000 znakova. Poznati su primjeri gdje su naše velike firme sa nekoliko tisuća radnika vodile svoje cjelokupno poslovanje na sistemima ne većim od 32K memorije.

b) Dimenzije miniračunala su zaista mini: 24×14×3 cm i težine od oko pola kilograma. Ono zaista izgleda kao igračka u usporedbi s ogromnim kabinetima koje su dizalice morale unositi u poslovne prostorije.

c) U usporedbi s ranijih nekoliko desetaka ili stotina tisuća dolara današnja cijena od niti 200 dolara zaista nije pretjerana.

Zbog smanjivanja cijena, računala su se sve više širila. Sve češće se čuje pitanje: »Što može raditi ovo računalo?«

Na ovo pitanje može se odgovoriti tek nakon razjašnjenja nekoliko osnovnih pitanja:

Osim ovih činilaca postoji još čitav niz drugih koji utječu na mogućnost stroja. U grubljoj varijanti računala možemo usporediti s automobilima koji se međusobno u mnogo čemu razlikuju i neprestano uspoređuju i testiraju. Da bi netko korisno upotrijebio računalo mora imati određene potrebe. Sigurno je da postoji veliki broj ljudi kojima računala uopće nisu potrebna, niti će ikada biti. Stvarne mogućnosti su vrlo šarolike i raznolike.

U svakom broju pokušat ćemo prikazati poneku praktičnu primjenu novih stanovnika naših domova. Ako ste i sami uspjeli pronaći neku zanimljivu primjenu, javite se!

Željko Pažur

Po struci sam inženjer elektrotehnike, a veliki hobi mi je zrakoplovno jedriličarstvo. U ovom članku opisat ću malo računalo koje sam zamislio i načinio sam i u kojem sam povezao oba ova moja područja interesa.

Zrakoplovno jedriličarstvo je sport sa više disciplina, od kojih je jedna od najatraktivnijih natjecanje u preletu, kojoj je cilj u što kraćem vremenu obletjeti zadane točke na terenu. Kako jedrilica nema vlastitog pogona, to je nakon povlačenja avionom na 300 do 500 m visine nužno koristiti uspone zračne struje da bi se izvršio zadani prelet. Moderne jedrilice standardnog razreda imaju omjer planiranja (finesu) od 1:30 do 1:40 što je omjer propadanja i pređenog puta. Zato je potrebno za svakih 100 km preleta dobiti u visini 2500 do 3000 m. Spomenimo još da su danas ostvareni letovi jedrilicama dulji od 1700 km.

Postoje određene meteorološke i aerodinamične zakonitosti koje određuju kojom brzinom treba letjeti između termičkih »stupova« (usponi strujanja), da bi nam putna brzina bila najveća. Određivanje te brzine dosta je složen pro-

Računalo upravlja jedrilicom

- Cijene miniračunala protežu se u rasponu od 40 do 3000 engleskih funti.
- Osim po cijeni, razlikuju se po veličini memorije i brzini procesora.
- Kvaliteta programske podrške također je bitna karakteristika računala.
- Jedna od bitnih osobina je i mogućnost priključenja na ostale periferne uređaje.

– brzini preskoka koja daje najveću putnu brzinu (brzini najbolje finese)
– veličini finese za tu brzinu
– doletu sa trenutne visine tom brzinom.

Određivanje visine

Kako je u preletu bitna visina prema aerodromu polijetanja odnosno slijetanja, odlučio sam određivati visinu pomoću Babinettove formule za barometričko određivanje visine koja ima oblik

$$h2 - h1 = 16000\ (1+0{,}0036604 \cdot Tsr) \cdot \frac{p1-p2}{p1+p2}$$

gdje je

h2 – h1 = relativna visina prema aerodromu u metrima

Tsr = srednja temperatura sloja zraka između h1 i h2 u °C. Aproksimiramo ju sa

$$Tsr = \frac{T1+T2}{2}$$ gdje su T1 i T2 temperature na nivou h1 odnosno h2.

p1 = pritisak zraka na nivou h1

p2 = pritisak zraka na nivou h2

Vidimo da nam je za ovo izračunavanje potrebna i temperatura kao ulazni parametar.

Brzinu penjanja dobijemo dijeljenjem relativne visine s vremenom iz vremenskog brojača

$$Vp = \frac{h2-h1}{\Delta t}$$

blem budući da ona ovisi o mnogim činiocima (parametrima). Jedrilica nema autopilota, pa je jedriličar cijelo vrijeme leta zauzet navigacijom i održavanjem elemenata leta. Zato sam odlučio načiniti uređaj koji će sam uzimati potrebne parametre i iz njih izračunavati najbolje brzine za prelijetanje između termičkih stupova. Elegantno rješenje za ovakav relativno složeni zadatak je korištenje mikroprocesora. Kvalitete jednom načinjenog uređaja mogu se popravljati usavršavanjem i izmjenama programa koji obavlja povjerenu mu zadaću, uglavnom bez zahvata u samu opremu.

Prvo je bilo potrebno definirati problem i postaviti matematički model uređaja. Ulazni parametri potrebni za rješenje problema su:

– visina, odnosno brzina penjanja
– brzina jedrilice (za preskok)
– težina jedrilice
– smjer i jačina vjetra

Obradom ovih parametara dobivamo podatke o:

Određivanje brzine

Brzinu jedrilice određujemo tako da nam razlika između statičkog i dinamičkog pritiska na Pitot ili Venturi cijevi služi kao modifikator adrese tabele u kojoj su pohranjeni podaci o brzini u ovisnosti o toj razlici pritisaka.

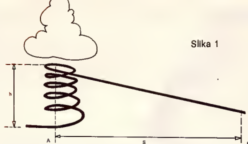
Težina jedrilice

Moderne jedrilice su opremljene tankovima za vodobalast. Povećanje težine jedrilice povećava njenu probojnost, to jest omogućuje postizanje boljih finesa pri većim brzinama. Informaciju o punom ili praznom tanku za vodobalast uređaj uzima sa prekidača, kojeg pilot postavlja na panelu.

Smjer i jačina vjetra

Ove parametre uzimamo također s prekidača smještenih na panelu uređaja, a postavlja ih također pilot.

Ovime smo definirali sve parametre potrebne za izračunavanje željenih vrijednosti, a karakteristike jedrilice čuvamo tabelarno zapisane u ROM-u uređaja. U preletu razlikujemo dva osnovna ciklusa koji se izmjenjuju: ciklus penjanja u kojem jedrilica obično kruži nad istim mjestom



Slika 1

i, koristeći se usponim strujanjima, dobiva na visini i ciklus preskoka u kojem jedrilica leti s vrha iskorištenog termičkog stupa prema podnožnju slijedećeg.

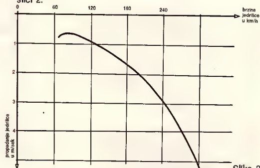
Zbog toga računamo dva slučaja:

a) U penjanju određujemo željene parametre na osnovi izmjerenog prosječnog dizanja i pretpostavljenih uvjeta u fazi preskoka.

b) U preskoku određujemo željene parametre na osnovi prethodnog prosječnog dizanja i izmjerenih uvjeta u fazi preskoka.

Uređaj sam razlikuje o kojem je ciklusu riječ, već prema tome da li trenutna visina raste ili pada.

Ovisnost brzine jedrilice i njenog vlastitog propadanja dana je dijagramom nazvanim POLARA BRZINA JEDRILICE. Karakterističan izgled jedne polare brzina dan je na slici 2.



Slika 2

Razmotrimo sada jedan ciklus penjanja i preskoka prema slici 1. Mjesta A i B udaljena su međusobno s metara. Ako nad mjestom A penjemo brzinom v1, onda ćemo se tu zadržati vrijeme $T1 = \frac{h}{v1}$ da bi se popeli za visinu h.

Nakon toga krećemo u preskok nekom brzinom v2 zbog koje propadamo brzinom w (vidi polaru brzine). Znači, ako želimo nad mjesto B doći na istoj visini na kojoj smo počeli penjati iznad mjesta A, treba nam $T2 = \frac{h}{w}$ vremena.

Ukupno vrijeme putovanja od mjesta A do mjesta B je $Tuk = T1 + T2$, odnosno $V\ putna = \frac{s}{Tuk}$. Ako polaru brzina aproksimiramo krivuljom drugog reda $av^2 + bv + c = w$ možemo naći:

a) maksimum funkcije putne brzine
b) maksimum funkcije finese

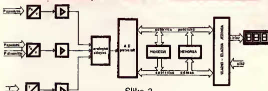
a odavde, uz pomoć polare brzina, i ostale željene parametre. Ako ovom problemu dodamo još i utjecaj vjetra, onda se on bitno komplicira i mikroprocesor bi trebao dosta vremena za izvršenje svih potrebnih proračuna. Zbog toga sam načinio kompromis: kompletne proračune optimalizacije za sve uvjete izveo sam na računalu UNIVAC 1100 u programskom jeziku FORTRAN, i te rezultate pohranio u ROM uređaja. Prikladnom organizacijom ROM-a omogućio sam da se na osnovu parametara (srednje dizanje, režim sa ili bez balasta, komponenta vjetra) izračunava adresa sa koje se uzima gotovi rezultat pripremljen na velikom sistemu. Takav način rada skratio je vrijeme potrebno za dobivanje rezultata nekoliko puta i pojednostavio strojni program. Ovime bi završio o matematičkoj osnovi cijelog problema, iako su neki pojmovi samo dotaknuti. Za detaljno rješenje problema optimalizacije preleta bilo bi potrebno napisati podeblju knjigu, a nama je bio cilj samo naznačiti kompleksnost problema koji treba riješiti.

Sada se postavlja pitanje kako riješiti postavljeni zadatak. Prvo, potrebni su pretvarači koji neelektrične veličine (apsolutni pritisak, diferencijalni pritisak i temperatura) pretvaraju u električne. Zatim trebamo te električne veličine prilagoditi da imaju povoljan hod signala dovoljan za pretvorbu analogno-digitalnim pretvaračem. Taj pretvarač pretvara električnu veličinu ulaznog signala u određenu binarnu kombinaciju koja se onda može dalje obrađivati. Pretvorba zahtijeva još i analognu sklopku koja propušta jednu od tri ulazne veličine na analogno-digitalni pretvarač. Digitalni dio uređaja organiziran je oko mikroprocesora koji upravlja čitavim uređajem na osnovi programa zapisanog u ROM-u. Ulazno-izlazni saobraćaj odvija se preko ulazno-izlazne jedinice. Prikaz rezultata je na troznamenkastom sedmodjelnom polju realiziranom upotrebom svjetlećih dioda (Seven segment LED display).

Ne upuštajući se u detaljne analize električnih svojstava sklopova koje ćemo koristiti možemo načiniti organizacionu blok skicu uređaja.

Popis sklopova korištenih u uređaju:

oznaka	
Lx1602A	pretvarač apsolutnog pritiska u napon
Lx1601D	pretvarač diferencijalnog pritiska u napon
AD590MF	pretvarač temperature u struju
AD522BD	diferencijalno pojačalo
AD741LN	operaciono pojačalo
DG190BP	analogna sklopka digitalno upravljana
AD574JD	analogno-digitalni pretvarač
MC6802	8-bitni mikroprocesor porodice 6800 sa 128 B RAM-a
MC6821	paralelni ulazno-izlazni sklop porodice 6800
2732	4K × 8 reprogramibilna memorija za čitanje (EPROM)
74LS154	demultipleksor iz 4 u 1 od 16
74LS74	dvostruki D bistabil
74LS04	šesterostruki invertor
9368PC	četverostruki D bistabil s dekoderom i sklopom za pogon 7-djelnih znamenaka realiziranih sa svjetlećim diodama.



Slika 3

Pretvarače pritiska tvrtke National Semiconductor nabavio sam u Münchenu nakon gotovo godinu dana neuspješnog dopisivanja sa njihovim zastupništvom Eltrade u Trstu. Oba pretvarača radila su bez greške.

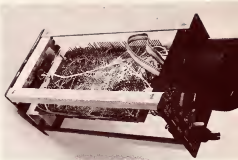
Analogno-digitalni pretvarač, pretvarač temperature u struju, diferencijalna i operaciona pojačala kupio sam poštom od tvrtke Analog Devices u Genevi. Proizvodi te tvrtke su vrhunski, ali im je takva i cijena.

Ostale sklopove kupio sam u Trstu u prodavaonici Electronic Shop.

Svaki pojedinačni sklop testirao sam prije ugrađivanja, tako da nisam imao problema sa sklopovima nakon sastavljanja uređaja.

Cijeli uređaj sačinjen je na pločici veličine Eurocard tehnikom WIRE WRAP-a. To je omogućilo veliku gustoću pakiranja (vidi sliku uređaja). Osim toga bio sam pošteđen razvoja komplicirane i skupe dvostruko tiskane pločice. Smatram da je WIRE WRAP tehnika pouzdana i zgodna za razvoj prototipova, jedino se prije početka rada mora te naoružati strpljenjem i imati dobre oči. Mana joj je što su podnožja relativno skupa, kao i alat i žica.

Glavni program složen je modularno. Sve matematičke operacije (izračunavanje h2-h1, množenje i dijeljenje u višestrukoj preciznosti) napisani su kao potprogrami koje poziva glavni program. Prije poziva potprograma glavni program postavlja parametre potrebne za izračunavanje u određeni dio memorije RAM-a koji onda koristi potprogram i opet ostavlja rezultat u dio RAM-a koji služi samo za vezu između glavnog programa i potprograma. Program sa potprogramom zauzima gotovo 4 kB ROM-a i još 4 kB ROM-a zauzimaju tabele sa izračunatim rezultatima i karakteristikama jedrilice. Primjera radi ovdje dajem odsječak programa koji upisuje brojku 8BB na display radi testiranja.



```
LDAA # $B0       »B« u gornja 4 bita, »O« je adresa jedinica
STAA DISABLE     zabrani upis
STAA DATAA2      upiši u izlaznu jedinicu
STAA ENABLE
STAA DISABLE     zaključaj u D bistabil s dekoderom
INCA             adresa slijedeće znamenke
STAA DATAA2      upiši u izlaznu jedinicu
STAA ENABLE
STAA DISABLE     zaključaj u D bistabil s dakoderom
INCA             adresa slijedeće znamenke
STAA DATAA2      upiši u izlaznu jedinicu
STAA ENABLE
STAA DISABLE     zaključaj u D bistabil s dekoderom
```

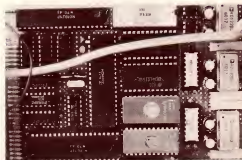
\#– direktno punjenje akumula tora vrijednošću

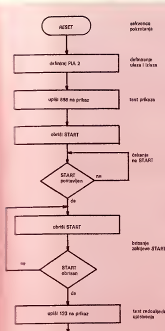
$– vrijednost je heksadecimalna

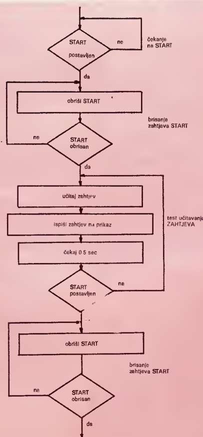
DATAA2 – simbolička adresa na ulazno-izlaznom sklopu

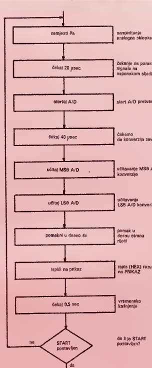
DISABLE – simbolička adresa koja preko birača sklopa aktivira zabranu upisa u D bistabil s dekoderom

ENABLE – simbolička adresa koja preko birača sklopa aktivira dozvolu upisa u D bistabil s dekoderom.

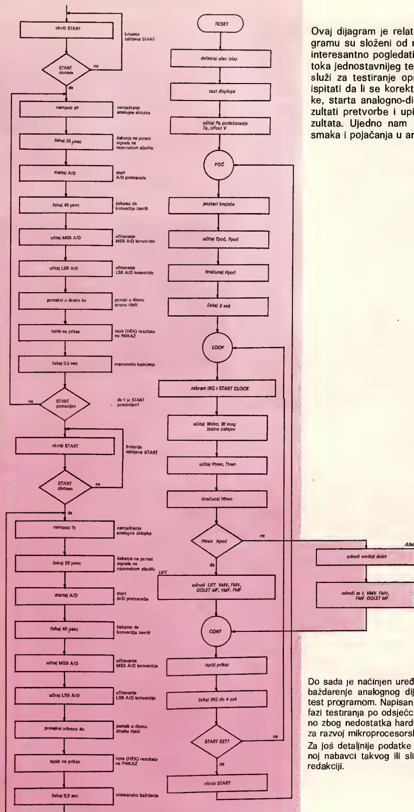








Ovaj dijagram je relativno grub. Neki blokovi u dijagramu su složeni od niza instrukcija. Zato je možda interesantno pogledati kako izgleda detaljni dijagram toka jednostavnijeg test programa za isti uređaj, koji služi za testiranje opreme. Pomoću njega trebamo ispitati da li se korektno postavljaju analogne sklopke, starta analogno-digitalni pretvarač, očitavaju rezultati pretvorbe i upisuju u sklopove za prikaz rezultata. Ujedno nam pomaže kod podešavanja posmaka i pojačanja u analognom dijelu uređaja.



Do sada je načinjen uređaj, izvršeno je testiranje opreme, baždarenje analognog dijela u baro-komori i testiranje s test programom. Napisan je i glavni program koji je sada u fazi testiranja po odsječcima. To je testiranje dosta otežano zbog nedostatka hardware-ske i software-ske podrške za razvoj mikroprocesorskih sistema kod nas.

Za još detaljnije podatke o uređaju i programu i eventualnoj nabavci takvog ili sličnog uređaja možete se obratiti redakciji.

Vladimir Sabljar

MALA RAČUNALA u svijetu i kod nas

Premda smo već potpuno navikli na sve kraće vrijeme potrebno od najave novog tehnološkog proizvoda do njegove široke primjene, elektronična računala doživljavaju u posljednjih dvadesetak godina tako nagli razvoj da su rijetki pojedinci koji ga uspijevaju pratiti u potpunosti. Iz dana u dan pojavljuju se nove stvari, a mnoge od njih donose značajne promjene u neposrednoj budućnosti. I tako, samo deset godina nakon Altaira, svatko tko za to ima interesa, može kupiti malo osobno računalo i iskoristiti ga za ono što mu treba. Na tržištu je, po relativno niskoj cijeni, lako pronaći program koji najbolje odgovara potrebama korisnika. Oni s većim prohtjevima također dolaze na svoj račun – modeli namijenjeni profesionalnom radu sve su češći u uredima i domovima, a ono što mogu učiniti, prije ne tako dugog vremena bilo je samo u domeni velikih računalskih centara. Mogućnosti osobnih računala rastu takvom brzinom da iznenađuju čak i stručnjake – vrlo ćete često i među njima pronaći one koji, premda teoretski znaju što mikroračunala danas mogu, na njih gledaju s prilično nepovjerenja. No, vrlo je malo pojedinaca koji nakon što sjednu za tastaturu malog računala ne priznaju da se radi o pomagalu koje će u slijedećim godinama izmijeniti način poslovanja i života uopće.



Dok sjedim za tastaturom svog računala koje mi je zamijenilo pisaći stroj, i u zelenim znakovima monitora pokušavam pronaći inspiraciju za priču o stanju kompjutorizacije (pritom se misli na nezadrživo širenje osobnih računala) u svijetu i kod nas, otkrivam prilično zgodnu koincidenciju. Vrijeme nastajanja i izlaska ovog časopisa poklopilo se s desetom godišnjicom vrlo značajnog događaja u povijesti računalske tehnologije. U siječnju 1975. jedan od najpopularnijih američkih hobističkih mjesečnika za elektroniku, Popular Electronics, na naslovnoj stranici objavljuje sliku prvog osobnog elektroničkog računala nazvanog Altair, s opširnim člankom o njegovim osobinama i mogućnostima primjene (premda su o tome u to »kameno doba« postojale tek vrlo zamagljene misli). Prema tome, trenutak je više nego prikladan da se priča o nastanku i razvoju osobnih računala, kao i svega što ih prati.

Premda entuzijasti kojima dugujemo današnje modele osobnih računala vjerojatno nisu slutili kamo će tehnologija u tako kratkom vremenu odvesti (mnogi upućeni, naime, tvrde da je njihov nastanak posljedica općeg razvoja, te da je »otkriće« mikroračunala naprosto »visjelo u zraku«), to ne umanjuje njihovu zaslugu. Prva polovica sedamdesetih godina doba je konačne potvrde vrijednosti elektroničkih računala, te vrijeme kad se tehnologija razvila dovoljno za uvođenje tzv. »minikompjutora« i stvaranje mikroprocesora, odnosno računala na jednom jedinom integriranom krugu, ili čipu, kako ga popularno nazivamo. Običan je smrtnik to doživio putem poplave džepnih računala, kalkulatora, koji su za kratko vrijeme osvojili svijet. Valja znati da oni počivaju na jednakim principima kao i mikroračunala, a razlika je u



gao prilično uspjeh: tokom nešto više od godine i pol prodano je nekoliko stotina primjeraka, no programi i BASIC prevodilac, koji su se upisivali s papirne vrpce, nisu bili baš najzgodniji medij. Tada se netko sjetio mogućnosti zapisivanja informacija na običnu kazetofonsku vrpcu, i u srpnju 1976. stiže SOL, nasljednik Altaira koji je imao sva potrebna poboljšanja – ugrađeni upravljački sistem, video pločicu, tastaturu (nije više bilo prekidača), i to sve u kutiji veličine pisaćeg stroja. No SOL nije imao većeg uspjeha i ubrzo nestaje s povijesne pozornice, a razlog je bio Apple I, računalo izrađeno na jednoj jedinoj štampanoj ploči, prethodnica sjajne budućnosti.

U to pionirsko doba, nove tvrtke nastaju kao gljive poslije kiše, no većina propada nakon kraćeg vremena. Samo oni koji uspijevaju zainteresirati financijere (a to nije nimalo lako), imaju nešto lakši san. Zanimljivo je da se povijest ponavlja – situacija u industriji mikroračunala doživljava danas slična previranja kao u počecima, premda su razlozi danas nešto drukčiji. Naime, nakon što su veći proizvođači prepoznali vrijednosti osobnih računala, u sve se ve-

jednoj jedinoj osobini – njih se, naime, nije moglo programirati, tako da se zapravo radilo o računalima specijalne namjene. (S obzirom na sve manje dimenzije pojedinih modela osobnih računala, danas je, naravno, mnogo teže povući granicu između njih i kalkulatora). Stvaranjem prvih mikroprocesora, a jedan od njih je bio i 4-bitni Intelov 4004, stvaraju se temelji nove industrije. Vrlo je zanimljiva činjenica da je proizvođačima računala i poluvodičkih elektroničnih elemenata bilo potrebno vrlo mnogo vremena da shvate što se događa. Sve što je učinjeno od 1972. do 1976. zasluga je pojedinaca, »amatera« – korisnika koji su spoznali mogućnosti novog medija. Slobodno bismo mogli reći da su osobna računala stvorena kao hobistički projekti u kupaonicama (slučaj Altaira), ili garažama (što je svima poznat primjer). Jedan od vizionara, praktični nepoznat svima osim nekolicini upućenih, je i Don Lancaster, u posljednje vrijeme sve češće nazivan i »ocem osobnih računala«. On je još 1973. stvorio tzv. TV-pisaći stroj, koji nije bio ništa drugo do teleprinter, tastaturom povezan s TV-ekranom, što je osnovni korak u stvaranju osobnog računala opće namjene. Sve dotada, a to možda i nije toliko poznato, s računalima se izravno komuciniralo uglavnom preko običnih sklopki na komandnoj ploči, a podaci su se uspisivali i ispisivali preko bušenih vrpci, kartica, ili na pisačima. Lancasterov TV-pisaći stroj bio je nešto poput današnjih računalskih terminala, i kao takav svakako prethodnik osobnih računala. No odlučujući korak morao je tek stići, a to je bio 8-bitni mikroprocesor koji dolazi na tržište početkom 1974. Intel je stvorio dvije vrste, 8008 koji nije doživio veći uspjeh, i 8080 koji postaje standardom industrije. Upravo na njemu se osnivao i Altair, za današnje pojmove primitivni uređaj koji se sastojao od kutije centralnog procesora na kojoj su bili prekidači za komunikaciju s računalom i teleprintera s čitačem papirne vrpce (to je bio jedini način unosa i ispisa podataka). Altair je posti-

ćem broju ubacuju u igru. Naravno, uspjeh na tržištu u prvo je vrijeme bio osiguran zbog gotovo nezasitne gladi potrošača, no s vremenom se želje i potrebe usklađuju, i svi oni, koji računaju samo s brzom zaradom, prije se ili kasnije moraju suočiti s bankrotom. Pratimo li stručne časopise, utvrdit ćemo da svakog mjeseca desetak, pa i više, relativno poznatih američkih hardverskih i softverskih tvrtki zatvara vrata. Možda je, stoga, ovaj trenutak prilično pogodan za proučavanje uspjeha (odnosno neuspjeha).

Mikroračunala su na tržište stigla u dva vala. Od uređaja prve generacije mogli bismo nabrojiti IMSAI, SOL, Apple II, PET, TRS-80 Model I, Cromemco, Texas Instruments 99/4, kao i nekolicinu kod nas nepoznatijih modela poput Vector Graphic, Northstar Horizon, Wave-Mate, te TDL. U drugom valu su stigli Atari, Commodore Vic-20 i 64, Apple IIe, IMB PC i čitav niz tzv. »prenosivih« računala, od kojih su svakako najpoznatija Osborne 1 i Kaypro II. Sad se nalazimo na pragu treće generacije koja će korisnicima omogućiti još jednostavniji rad nego do sada. Njeni su predstavnici računala s 32-bitnim procesorima, poput Apple Macintosha.

Odredimo li ključne osobine uspješnih modela, to će nas s vrlo velikim stupnjem sigurnosti voditi prema smjernicama koje osiguravaju dobru prođu na tržištu, a to drugim riječima znači da su po volji prosječnom kupcu. Između elemenata koje valja proučiti svakako su mogućnosti proširivanja, vrsta vanjske memorije, slobodna mjesta za dodatne pločice, grafika visokog razlučivanja i grafika u boji, interno stvoreni video signal, a važna je, naravno, i cijena. Prema tome, neće nas mnogo začuditi kad utvrdimo da dva najuspješnija modela do danas, Apple II i IBM PC, imaju mnogo zajedničkog: mogućnost grafike u boji, slobodna mjesta za proširivanje, disk jedinice promjera 5,25 inča, te memorijski organiziran, interno stvoreni video signal. No pođemo li dalje, vidjet

ćemo da slične osobine ima većina modela na tržištu, pa čak i oni koji su manje ili više neslavno propali. **Stvarni zajednički nazivnik uspjeha** – osim činjenice da svako računalo koje želi bolju prođu mora imati tastaturu i disk jedinicu – su entuzijazam amatera i programska podrška nezavisnih proizvođača. Koristimo li taj kriterij kao mjeru uspjeha ili neuspjeha, ustanovit ćemo da praktički ne možemo pogriješiti. To je i glavni razlog zbog kojeg se Morrow ili Kaypro modeli prodaju tako dobro premda se osnivaju na procesoru Z80 koji je daleko od savršenstva. No Z80 omogućuje izvođenje svih CP/M programa, a njih je tokom godina sakupljeno stvarno vrlo mnogo. Koliki bi uspjeh ta računala postigla da su se oslanjala na vlastiti upravljački sistem koji nisu prihvatili amateri i stvaraoci programa?

Ovim posljednjim načeli smo temu koja je stvarni pokazatelj uspjeha kompjutorizacije. Naime, nije teško utvrditi da umijeće programiranja daleko zaostaje za napretkom tehnologije. I tako, premda na tržište neprekidno stižu novi modeli sve moćnijih računala, programi koji ih moraju oživjeti tek kaskaju za njima. U posljednje vrijeme ima mnogo primjera za takav slijed događaja – dovoljno je spomenuti, na primjer, Sinclaira QL koji je najavljivan na velika zvona, no ubrzo se ustanovilo da pati od mnogih nedostataka u upravljačkom sistemu, pa mu čak i BASIC prevodilac ne radi kako treba. Slične se stvari događaju češće nego što bismo mislili, a glavni je razlog potcjenjivanje vremena potrebnog za stvaranje dobrog programa (bez obzira na vrstu i namjenu). Još jedan zgodan primjer je i CAI/CAL (Computer Aided Instruction/Computer Aided Learning – podučavanje i učenje uz pomoć računala). Naime, to područje je jedno od kojih bi čovječanstvo u cjelini imalo najveće koristi. No čak i u Sjedinjenim Državama gdje je

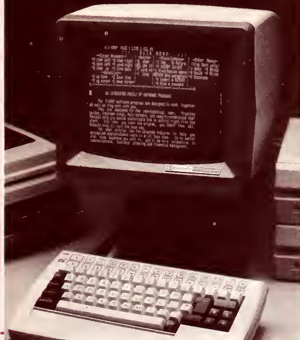


teško govoriti o nedostatku programera, još uvijek nije stvoreno dovoljno kvalitetnih obrazovnih programa. Ne samo što je to težak posao, već i proces učenja nije još uvijek dovoljno poznat. No, osobna će računala pomoći i na tom području; čim se u rad uključi dovoljan broj nastavnika, rezultati će iz dana u dan biti bolji. Nije stoga čudno da se svakih pet godina očekuje podvostručavanje broja računala u obrazovnim institucijama (danas ih u SAD ima gotovo 400 tisuća, deset puta više nego prije samo tri godine).



Korištenje 16 i 32-bitnih mikroprocesora, te sve niža cijena glavnih i perifernih memorija vodi stvaranju računala čija se mogućnost obrade podataka počinje uspoređivati s velikim sistemima. Naravno, koliko će uspjeha u tome biti, prvenstveno ovisi o programima. Zbog radnih memorija velikog kapaciteta, u posljednje su vrijeme sve popularniji tzv. »integrirani« paketi, u kojima se obično nađe program za obradu teksta, pohranjivanje i organizaciju podataka, te proračunavanje (spreadsheet). Nakon što se takav »paket« učita s diska, programi nam neprekidno stoje na raspolaganju, a kod nekih modela ih čak možemo koristiti istovremeno. Naravno, njihova je cijena prilično visoka, i zato su primamljivi za svakog proizvođača. Koliko su potrebni korisnicima, to je sasvim drugo pitanje, no programere to (barem isprva) ne brine. Nažalost, vrlo je teško pronaći »paket« ujednačene kvalitete, pa se s dobrim programom za obradu teksta mora kupiti inferiorna baza podataka, ili neka druga neželjena kombinacija. S druge strane, možda će ovo razdoblje računalske povijesti trajati i kraće nego što smo mislili: IBM za svoj PC sprema upravljački sistem koji će korisniku omogućiti da »integrira« programe koje želi, odnosno one koje već posjeduje i na koje je navikao. Time je svakako načinjen jedan od prvih koraka prema potvrdi činjenice da odluku o tome kako će se računalo koristiti valja prepustiti vlasniku, a ne (prečesto nestvarnim) zamislima nekog programera. Valja se nadati da će takav slijed događaja u budućnosti biti sve češći.

A gdje se u čitavoj toj priči nalazimo mi? Sudeći po onome što se događa u posljednje vrijeme, mogli bismo reći da se konačno nešto kreće. Poslije nekoliko godina u kojima se naša kompjutorizacija osnivala isključivo na nezakonitom unosu opreme iz inozemstva, danas se u trgovinama može pronaći nekoliko kućnih modela (Sinclairov Spectrum, Commodore 64. . .). S obzirom na iskustva razvijenih zema-



lja gdje se pokazalo da se rezultati ne mogu postići bez vrlo široke osnove (što drugim riječima znači bez jeftinih računala), za očekivanje je da će se tokom slijedećih par godina svijest o korisnosti osobnih računala dovoljno proširiti za sustavniju akciju.

To, doduše, ne znači da je Jugoslavija u mikroračunalskom smislu potpuno »djevičanski« teritorij. Pokušaji da se nešto više učini sežu prilično daleko u prošlost. Još početkom prošlog desetljeća ustanovljen je u Zagrebu Multimedijski centar sa zadatkom da (uglavnom) mlade nauči računalskoj pismenosti. Prvi su se modeli osobnih računala našli u njemu vrlo kratko vrijeme nakon pojave na zapadu, i do danas su njima naučili rukovati deseci tisuća mladih. Nažalost, industriji je trebalo mnogo vremena da se uklopi u nove trendove, trebalo je načiniti sve pogreške koje su drugi učinili par godina prije nas, i tek se u posljednje vrijeme opažaju značajnije promjene (u najskorije vrijeme nekoliko proizvođača namjerava započeti prodaju mikroračunala različite namjene). Sve što je prije učinjeno nije otišlo mnogo dalje od prototipova i vrlo ograničenih serija (primjer Iskrinog Partnera, Velebitovog Galeba; itd.), a veći je uspjeh postigao samo kućni model Galaksija. Razlog tome je, naravno, cijena: druga su računala na našem tržištu, općenito govoreći, precijenjena barem pet do deset, pa i više puta. Nije, stoga, čudo da u našim uvjetima nije lako pronaći kupca. Pojava većeg broja modela koji se mogu nabaviti po konkurentnim cijenama (bez obzira da li za dinare ili u konsignacijskoj prodaji) mogla bi utjecati i na naše proizvođače. Samo će masovnost omogućiti našoj zemlji da započne smanjivanje »kompjutorskog jaza«, zaostatka koji nas s prolaskom vremena sve skuplje stoji.

S obzirom na vrlo specifične prilike, jedan od prvih zadataka mikroračunala kod nas bit će prosvjećivanje. Zato i ne čudi što se danas mnogo govori o akciji »1000 računala u škole«, no da bi ona u potpunosti uspjela, morat će se učiniti još mnogo toga. Najvažnija od svega je kvalitetna obuka nastavničkog kadra, kako bi na najbolji način znao iskoristiti mogućnosti novog medija, kao i stvaranje dovoljno velike biblioteke obrazovnih programa. Pogriješit ćemo ako sve budemo htjeli načiniti sami, jer ćemo tako izgubiti još više dragocjenog vremena, a to ni na koji način ne smijemo dopustiti.

Ruđer Jeny

»Prosvjetin« računalski happening

Jedan od pokušaja popularizacije malih računala učinila je i RO »Prosvjeta« koja pokretanjem biblioteke »Popularna informatika« (dosad izašao BASIC Jure Špiera, u pripremi komplet od 4 knjige engleskih autora »Svijet osobnih računala«) pokušava reagirati na sve veće čitalačke potrebe na tom području.

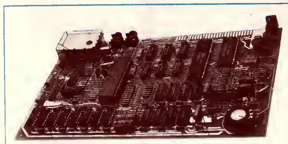
U sklopu tih nastojanja upriličen je i mali računalski »happening« u zagrebačkoj knjižari »Prosvjeta«, Trg bratstva i jedinstva 5. Demonstracija programske kazete »Radio-Študenta« iz Ljubljane za ZX Spectrum prerasla je u mali festival informatike, događaj koji je zahvaljujući dobroj reklami privukao veliki broj namjernika. Predvođena R. Jeny-jem, ekipa od nekoliko demonstratora je na dva ZX Spectruma i jednom Apple II računalu, prikazala neke od mogućnosti ovog suvremenog informatičnog medija. Osim toga, svi su mogli dobiti i savjet o problemu koji ih zanima. Demonstracija (uz prikazivanje BBC-jeve serije »Kako najbolje iskoristiti osobno računalo«) je trajala 5 dana, od 21. do 27. studenog. S obzirom na uspjeh, za očekivati je da će se nešto slično ponoviti u najskorijoj budućnosti.

Žarko TARAŠ

CAD/CAM – sadašnjost i budućnost industrije



ZX Spectrum +



Zatišje koje je nastupilo nakon pojave QL-a konačno je prekinuto: pred nama je najnoviji proizvod Sinclaira, ZX Spectrum+. Mnogi će vjerojatno ostati razočarani, jer očito je da se ne radi o novom proizvodu već samo o usavršenoj verziji postojećeg računala; međutim ova vijest će nesumnjivo obradovati sve korisnike »starih« Spectruma, jer ujedno znači da se izuzetno uspješan razvoj njihovih sistema nastavlja.

Da naglasimo odmah na početku, novi Spectrum je posve prilagodljiv (*compatible*) sadašnjem modelu, bitna razlika je u izvedbi tastature koja je, pored mehaničkog poboljšanja (kao *QL*) također proširena dodatnim tipkama. Uvažavajući vjerojatno rješenja raznih nezavisnih proizvođača tastatura za ZX Spectrum, te vlastita iskustva dobivena pri plasmanu i upotrebi QL-a; Clive Sinclair je, za svoj sigurno najpopularniji model, ponudio kompromisno rješenje. Umjesto dosadašnjih 40 »gumica« sa raznobojnim oznakama funkcija, Spectrum+ ima čak 58 tipki, što uključuje i podužu (ali ne standardnu) razmaknicu (*SPACE BAR*). Boja tastature je i dalje crna, dok su sve oznake bijele, pa je sličnost sa QL-om očita. Sve funkcije pojedine tipke ispisane su na samoj tipki, što početniku može zadavati problema (pored osnovnog znaka-slova tipke sadrže i do četiri različite funkcije), no stari korisnici su već naviknuti na ovu manu. Raspored osnovnih funkcija je uglavnom isti kao i na standardnim modelima. Dodatne tipke u odnosu na »stari« Spectrum su:

- *TRUE VIDEO I INV VIDEO* (na početku prvog reda, ispred »1«),
- *BREAK* (na kraju prvog reda, iza »∅),
- *DELETE i GRAPH* (na početku drugog reda, ispred »Q«),
- *EXTEND MODE I EDIT* (na početku trećeg reda, ispred »A«),
- *ENTER* (velika »L« oblikovana tipka na kraju trećeg reda, iza »L«),
- *CAPS SHIFT* (široke tipke na početku i kraju četvrtog reda),
- *CAPS LOCK* i ».« (također na početku i kraju četvrtog reda),
- *SYMBOL SHIFT:* (dvije tipke, na početku i kraju dodatnog, petog, reda koji još sadrži tipke: ";", """", ",", 4 tipke pokazivač za (cursor), te nestandardno »usku« razmaknicu.

Najkorisnija od navedenih unaprijeđenja tastature je svakako izvođenje posebnih tipki za interpunkciju. Ljubitelji igara će svakako pozdraviti izdvojene tipke za pokazivač, dok programerima nisu na odmet posebne tipke za pisanje (*EDIT*), brisanje (*DELETE*), BREAK, EXTENDED MODE itd. Sve u svemu, novi model pruža potencijalnim korisnicima u svakom pogledu »ugodniji« rad, no eventualna profesionalna primjena zahtijevala bi ipak više. »INES« tastatura za ZX Spectrum sa čak 83 (!) tipke, najbolja je potvrda ove tvrdnje. S druge strane nova verzija Spectruma (postolje s oznakom ISSUE 3B) znači da kvalitet računala i nadalje osigurava njegovu dobru prodaju, odnosno da nekoliko miliona proizvedenih (i vjerojatno prodanih) komada, uz obilje programa predstavlja značajni potencijal koji je solidna osnova za dalje unapređivanje i razvoj.

Situacija u kojoj »stari« model nadvladava »novi«, a sa kojom se sada susrećemo promatrajući Sinclairov proizvodni program, odnosno odnos proizvoda ZX Spectruma+ prema novom QL-u izuzetno je rijetka u modernoj ekonomiji. Pri tom je jasno da tek izuzetni kvalitet može odoljeti nadolazećim, redovito tehnološkim unapređenjima.

Zbog toga bez straha nabavite ZX Spectrum ili ZX Spectrum+. U svakom slučaju radi se o dobrim računalima i vaša očekivanja će biti posve ispunjena, ako ne i premašena!

Damir Fio



ŠTO TREBA ZNATI PRI KUPOVINI »MIKRIĆA«



O načinima nabavljanja ili kupovanja mikroračunala nije potrebno previše pisati ili govoriti. Poznate su nam muke i napori koji su potrebni da bismo stroj prenijeli preko kućnog praga. »Kamo odlaze stara računala? Da li oni poput slonova i vojnika odjednom iščeznu bez traga? Da li je kupnja polovnog i jeftinog stroja dobar poslovni potez? Može li se nešto na računalu pokvariti?«

Ovakav tekst preveden iz stranog tiska ne uzbuđuje odviše našeg čitaoca. Još nismo došli do stupnja kad bismo razlučivali novo od starog jer je za nas gotovo sve novo. No, u svakom slučaju, gotovo svi uređaji su fizički zaista novi. Tehnološka zastarjelost još nije činilac koji vlasnike tjera na promjenu. Unatoč svemu, u zadnjih nekoliko mjeseci svjedoci smo velikih promjena u našim izlozima. Kupovina mikroračunala na konsignaciji više nije nemoguća. Moguća je čak i zakonita kupovina za dinare, u dućanima. Točno je, nažalost, da je oprema vrlo skupa, ali je napredak očit.

Ipak najveći broj osobnih računala još uvijek mijenja vlasnika u izravnoj trgovini tj. iz ruke u ruku. Pokušat ćemo opisati osnovne karakteristike koje je potrebno razmotriti pri takvoj kupovini:

- *ne treba kupovati opremu stariju od 2 godine,*
- *u našim uvjetima teško je povjerovati da postoji fizički istrošena oprema,*
- *izuzetak je tastatura koja može biti u lošem stanju ukoliko su se tipke upotrebljavale u svrhu igranja (umjesto palica) često, i na istim mjestima,*
- *kontakti na žicama su osjetljivi isto kao i elektromehanički dijelovi: diskete, štampači, ugrađeni kazetofoni,*
- *integrirani krugovi koji sačinjavaju memoriju stabilni su i pouzdani što i nije čudno kad se zna da ih proizvodi svega nekoliko proizvođača i da nisu skloni kvarenju,*
- *vlasnici bi trebali imati nekoliko »godina mira« (procjena je sedam). Nakon tog razdoblja počinju popuštati spojevi i dolazi do ulaženja vode i korozije koja nagriza spojeve čipova.*

Pri kupovini rabljenog računala najvažnije je ocijeniti razloge prodaje. Oni mogu biti raznoliki, ali se mogu razvrstati u jednu od grupa:

1. *ODUSTAJANJE = Vlasnik se malo poigrao, ali je zaključio da je BASIC ipak pretežak, igranja mu je dosta, nema djece, a za posao mu ne treba.*
2. *KUPNJA NOVOG = Vlasnik smatra da mu treba »nešto jače ili slabije«. Određen broj ljudi vrlo brzo nadiđe razinu mogućnosti svog računala i želi kupiti moćniji i inteligentniji model. Ima i vlasnika koji su kupili jače sisteme, misleći da će se njihovim iskorištenjem obogatiti, ali ubrzo uviđaju da im računalo stoji neupotrebljeno; prodaju ga i kupuju nešto jeftinije.*
3. *ZBOG ZARADE = Vlasnik je »prošvercao« računalo jedino i isključivo u namjeri da ga preproda.*
4. *ZASTARJELOST = Neka računala se uspijevaju održati duže vremena, dok neka iščeznu za vrlo kratko vrijeme. Neka se napuštaju i izbacuju iz proizvodnog programa jer se lansiraju novi proizvodi.*
5. *NEIPRAVNOST = Vlasnik prodaje računalo jer je neispravno. Vrlo često neispravnost nije odmah zamjetljiva, već se pojavljuje nakon nekog vremenskog razdoblja.*

Kupovina računala koje nije novo ima i svoju prednost. Mnogi proizvođači isporučuju prve pošiljke u kojima se nalaze neotkrivene programske greške ili čak neki dijelovi opreme nisu potpuno ispravni. Zato je ponekad bolje pričekati drugu seriju istog računala. »Demonstracioni« prikaz može se dosta razlikovati od stvarnih mogućnosti. Sve brojnija je literatura koja objavljuje preglede, vrednovanja i usporedbe pojedinih računala na osnovu kojih se može donijeti ispravna odluka o kupovini odgovarajućeg. Cijena je svakako činilac za i protiv. No nju treba sagledavati u odnosu na kvalitetu koju dobivamo. U nekim pregledima ocjenjuje se i vrijednost u odnosu na mogućnost: koliko smo dobili za svoje novce.

Važno je pripaziti i na postojanje odgovarajuće literature na jeziku koji poznamo. Posebno treba pripaziti kod kupnje opreme koja je rjeđa i starija jer ćemo literaturu teže pronaći. Sličan je problem i s programskom podrškom starijih i kod nas rjeđih serija.

Željko Pažur

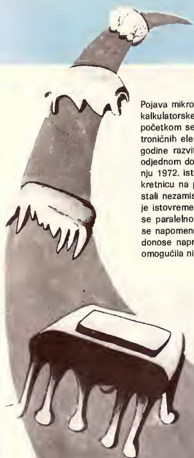
TEST ISPRAVNOSTI

Ovaj program potrebno je ukucati u Spectrum i zatim ga izvesti. Ostale upute za ovaj program nalaze se u Spectrumovoj biblioteci programa.

```
1 DATA 42, 101, 92, 68, 77, 62, 0, 1
19, 126, 254, 0, 192, 62, 255, 119, 126,
254, 255, 192, 167, 35, 237, 114, 200, 1
67, 237,122, 3, 24 231: FOR n=23296
TO 23325: READ m: POKEn,m: NEX
T n: PRINT USR 23296
```

lista programa MR-TEST

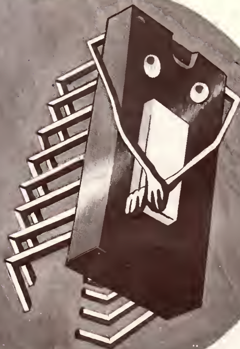
EVOLUCIJA MIKROPROCESORA



Pojava mikroprocesora, koji su po svojoj strukturi značili napredak u odnosu na dotadašnje kalkulatorske sklopove, uvjetovana je pojavom tehnike visokog stupnja integracije (VSI) početkom sedamdesetih godina. Razvoj VSI tehnologije omogućio je povećanje broja elektroničnih elemenata na jediničnu površinu silicijske pločice. Tvrtka INTEL uspjela je 1971. godine razviti prvi mikroprocesor 4004 koji je sadržavao oko 3000 tranzistora i mogao odjednom dohvatiti i obraditi količinu informacija dugačku 4 bita. Ubrzo nakon toga, u travnju 1972. isti proizvođač ponudio je prvi osmobitni mikroprocesor 8008, koji je značio prekretnicu na području digitalne elektronike, jer su od tada mnogi elektronični sklopovi postali nezamislivi bez primjene mikroprocesorske tehnologije. Razvoj mikroprocesora pratio je istovremeni razvoj sklopova, kao što su ispisna (ROM) i radna (RAM) memorija, te su se paralelno pojavljivala i prva računala ostvarena u mikroprocesorskoj tehnologiji. Mora se napomenuti da sa stajališta arhitekture centralne procesne jedinice mikroprocesori ne donose napredak. Njihova je struktura vrlo jednostavna, ali je velikoserijska proizvodnja omogućila niske cijene, a time i širok prodor u najrazličitija područja ljudske djelatnosti.

Prvi mikroprocesori razvijeni su u tzv. P-kanalnoj MOS (PMOS) tehnologiji, koja nije omogućavala veliki broj integriranih elemenata na jednoj pločici i bila je relativno spora, te je ubrzo zamijenjena mnogo bržom i efikasnijom N-kanalnom MOS (NMOS) tehnologijom.

INTEL je i ovdje bio prvi sa svojim 8080 osmobitnim mikroprocesorom kojim je u stvari i započela mikroprocesorska revolucija. Ubrzo su se pojavili i drugi proizvođači, od kojih je najznačajnija pojava Motorolinog 6800 mikroprocesora, čija je velika prednost: samo jedan izvor napona.



Od čega se sastoji mikroprocesor? Najvažniji dijelovi su aritmetičko-logička jedinica za obavljanje računskih operacija, instrukcioni sklop koji služi za tumačenje instrukcija, memorijski adresni registar koji se brine za dohvat instrukcija i podataka iz memorije, skup registara za jednostavne radnje sa podacima, ulazno-izlazni pristupni sklopovi, te upravljačka jedinica koja generira interne i eksterne upravljačke signale. Podaci se između pojedinih dijelova vode paralelno po linijama odnosno sabirnici podataka (data bus).

S »vanjskim svijetom« mikroprocesor je povezan vanjskom sabirnicom podataka, adresnom sabirnicom, address bus: koja služi za naslovljavanje (adresiranje) memorijskog ili perifernog prostora i upravljačkim signalima koji služe za usklađivanje rada mikroprocesora i ostalih priključenih mu sklopova. Osnovni parametri koji karakteriziraju mikroprocesor jesu:

- dužina podataka u bitovima koji se odjednom mogu dohvatiti i obraditi
- maksimalni adresibilni memorijski prostor
- raspoloživi skup instrukcija
- maksimalna brzina (clock) na kojoj mikroprocesor može raditi.

Instrukcioni se kodovi i podaci iz memorije i preko sabirnice podataka pohranjuju u registrima mikroprocesora. Ukoliko je dužina podataka osmobitna kao što je slučaj kod osmobitnih mikroprocesora, broj raspoloživih instrukcija, ako se dohvat instrukcije obavlja u jednom koraku, ograničen je na 256. Isto tako se i podaci mogu dohvatiti jedino u osmobitnim blokovima. Ograničeni skup instrukcija omogućuje osmobitnim mikroprocesorima obavljanje samo jednostavnijih zadataka, kao što su prijenos podataka između registara, prijenos podataka u/iz memorije i perifernih sklopova, programske skokove, osnovne aritmetičke operacije na cjelobrojnim podacima i neke sistemske funkcije kao što je rad sa stogom (stack).

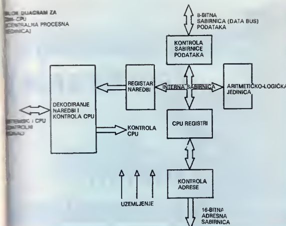
Proizvođači su ubrzo bili svjesni ograničavajućih mogućnosti dotadašnjih mikroprocesora pa su se pojavila tri nova osmobitna mikroprocesora INTEL-ov 8085, MOS Techov 6502 i Zilogov Z80.

Može se reći da je proizvodnja osmobitnih mikroprocesora kulminirala pojavom Z80 mikroprocesora, iako je i 6502 bio i ostao vrlo popularan zahvaljujući dobrom instrukcionom skupu i raznovrsnom načinu adresiranja, kao i tome što su ga u svoja računala ugrađivali takvi proizvođači kao što su Apple, Commodore i Atari. Zilogov Z80 u sebi uključuje (tj. emulira) glavna svojstva i instrukcije mikroprocesora INTEL 8080, ali posjeduje i prošireni skup instrukcija, dodatni skup registara kao i poseban sklop koji služi u postupku obnove sadržaja dinamičke memorije. Posebnu popularnost ovom mikroprocesoru dao je prvi standardni upravljački sistem za mikroprocesore CP/M s obimnom programskom podrškom, kao i sve šira primjena tog mikroprocesora na velikom tržištu kućnih računala, te najnoviji japansko-američki MSX standard.

Iako će se osmobitni mikroprocesori još zadržati na svjetskom tržištu, ipak moramo konstatirati da njihovo vrijeme prolazi. Šesnaestobitni mikroprocesori pojavili su se još 1976. godine kad je Texas Instruments proizveo 9900 mikroprocesora. Šesnaestobitni mikroprocesori nudili su širi i jači skup instrukcija, raznovrsne načine adresiranja u znatno povečanom adresibilnom memorijskom prostoru, napredniju arhitekturu kao i ugrađene sklopove za rad s brojevima u pomičnom brojevnom zarezu (floating-pont).

Dvije godine kasnije je INTEL stavio na tržište mikroprocesore 8086 i 8088. Mikroprocesori su programski međusobno primjenjivi (compatible) i kao podskup sadrže instrukcioni set 8080 mikroprocesora, jedino 8088 ima osmobitnu sabirnicu podataka što mu omogućuje korištenje osmobitnih memorijskih i perifernih sklopova, a time jeftiniju izgradnju računala. Oba mikroprocesora imaju dvadesetobitnu adresnu sabirnicu, odnosno mogu nasloviti 1,048.576 okteta. I jedan i drugi mogu se upotrijebiti kao glavni (master) procesor u mikroprocesorskom sustavu koji sačinjavaju još aritmetički koprocesor 8087, periferni procesor 8089 i sklop s ugrađenim upravljačkim sustavom 80130. Koprocesor 8087, prema naredbama koje dobiva od glavnog procesora, sklopovski vrši aritmetičke operacije nad brojevima u pomičnom brojevnom zarezu, dok periferni procesor samostalno obavlja izmjenu podataka s ulazno izlaznim napravama. Sklop 80130 ima ugrađenu programsku podršku, s oko 35 osnovnih jezgrenih procedura (kernel procedures), koje čine osnovu upravljačkog sustava. Takav višeprocesorski sustav omogućuje izgradnju višezadačnih i višeprogramskih (multitasking, multiprocessing) računala.





Mikroprocesor 8086 poslužio je kao osnova za izgradnju dvaju novijih INTEL-ovih šesnaestobitnih mikroprocesora 80186 i 80286. Oni sadrže kao osnovu 8086, zajedno s još nizom dodatnih sklopova u zajedničkom kućištu kao što su sklop za obradu prekida, vremenska brojila i DMA (direct memory access) sklopovi. 80286 pored ostalog sadrži i sklop za memorijsko poslovanje (memory management unit) kojo omogućuje rad s virtuelnom memorijom (virtuelni adresibilni prostor preslikava u fizički adresni prostor). Mikroprocesori 80186 i 80286 su također primjenjivi sa svojim prethodnicima. Snaga 80286 mikroprocesora posebno dolazi do izražaja u višekorisničkim (multi user) sistemima, a osim matematičkog koprocesora 80287 može koristiti i 80258 Ethernet koprocesor i 802370 tekstualno-grafički koprocesor.

Poslije velikog uspjeha mikroprocesora Z80, svi su očekivali da će Zilog napraviti i uspješan šesnaestobitni mikroprocesor. U neku ruku Z8000 to i jest sa svojom šesnaestobitnom internom i eksternom arhitekturom i mogućnošću adresiranja 8MB (megabyte) memorijskog prostora, ali su konstruktori pogriješili što novi i snažniji mikroprocesor nije programski primjenjiv sa svojim prethodnikom Z80 i time ga u »startu« ostavili bez programske podrške. Također nije napravljena verzija s osmobitnom sabirnicom podataka, što se također pokazala kao promašaj (8088 se bolje prodavao nego 8086!).

Z8000 dolazi u dvije verzije. Izvedba Z8001 može izravno nasloviti 8MB dok se ukupni naslovni prostor procesora Z8010 može proširiti do 48MB. U novije vrijeme proizvode se i Z8108 i Z8208 s osmobitnom sabirnicom, odnosno Z8116 i Z8216 sa šesnaestobitnom sabirnicom, koji su programski primjenjivi sa Z80. Jedna od posebnosti arhitekture Z8000 mikroprocesora jest naredbeni cjevovod (pipeline) u kome se vrši dohvat slijedeće instrukcije, dok se još izvršava prethodna, čime se ubrzava rad mikroprocesora.

U slijedećem broju bit će riječi o mikroprocesoru MC 68000 i drugim novostima.

Tomislav Žganec

Razvrstavanje (sortiranje) na malim računalima

1. UVOD

Pojam razvrstavanja podataka je manje ili više poznat svima koji se bave obradom podataka. To je nezaobilazan pomoćni program na većim računalima i služi za pripremu podataka za razne vrste obrada. (Npr. za usporedbe, listanja, kreiranje novih datoteka itd.).

Povećanjem procesne snage malih računala (brzi i kvalitetniji mikroprocesori, povećanje memorije, mogućnost korištenja raznih vanjskih uređaja za pohranu podataka), omogućuje da se osim raznih igara, na »kućnim« računalima mogu izvoditi i obrade podataka drugog tipa poput vođenja kućnog budžeta, kućne knjižnice, vođenje adresara, te mnogo drugih obrada koje proizlaze iz različitog djelokruga rada vlasnika računala. Kod obrada takvog tipa, javlja se, prije ili kasnije potreba stavljanja podataka u određeni redoslijed (npr. prikaz podataka knjižnice po abecednom slijedu naslova knjiga). U tom trenutku imamo potrebu za svrstavanjem podataka. Razvrstavanje (stavljanje u slijed po nekom kriteriju) podataka moguće je sa numeričkim ili nenumeričkim podacima zbog načina na koji računalo tretira svaki znak u memoriji. Za brojeve znamo da je jedan broj manji, jednak odnosno veći od drugo broja, prema opće poznatim matematičkim pravilima. Računalo ima mogućnosti da prema tim poznatim pravilima uspoređuje 2 broja. Kako računalo može uspoređivati znakove koji nisu numerički, npr. slova abecede, znaci "?", "!", "", ...? Svaki znak koji se nalazi u memoriji ima svoj određeni binarni zapis, a prema dogovoru, svakom znaku je određena numerička vrijednost, prema kojoj je stavljen u ispravan numerički redoslijed u odnosu na druge znakove. Da bi se olakšala i ujednačila obrada podataka, upotrebljava se međunarodni standard za prikaz znakova (imamo nekoliko standardnih prikaza, a najčešće se koristi ASCII).

U ovom članku govorit ćemo o nekim metodama razvrstavanja podataka u memoriji. Cilj nam je da pokažemo kako različite metode utječu na vrijeme trajanja razvrstavanja i tima na brzinu obrade podataka. Sva testiranja su rađena na računalu ZX Spectrum 48K, sa standardnim BASIC programskim tumačem (interpreter).

2. JEDNOSTAVNI SORT

U engleskoj literaturi se za ovu metodu razvrstavanja kaže »BUBBLE SORT« ili »SHUTTLE SORT« zbog stalne izmjene mjesta elemenata tabele koja se sortira. Da bismo lakše razumjeli ovaj način razvrstavanja, pogledajmo kako to izgleda na jednostavnom primjeru. Imamo tabelu od 5 brojeva koju želimo urediti po rastućoj vrijednosti brojeva:

17	12	41	9	22

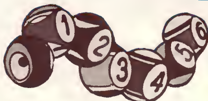
Sl. 1

Uspoređivanje počinjemo prvim brojem (17) i usporedimo ga s njegovim prvim desnim susjedom (12). Ako je prvi broj veći od drugog, zamijenimo im mjesta:

12	17	41	9	22

Sl. 2 Tabela nakon prve izmjene mjesta

Usporedbu nastavljamo tako da drugi broj usporedimo sa slijedećim s desne strane (41). U ovom slučaju je 41 veći od 17. Izmjena mjesta nije potrebna, a ispitivanje nastavljamo trećim brojem (41) koji uspoređujemo sa slijedećim brojem (9). U ovom slučaju rezultat usporedbe je takav da je potrebna izmjena mjesta:



```
12  17  9  41  22
```

Sl. 3

Prvi prolaz kroz tabelu završavamo ispitivanjem brojeva 41 i 22. I ovdje je potrebna izmjena mjesta, tako da je konačan izgled tabele nakon prvog prolaza:

```
12  17  9  22  41
```

Sl. 4

Vidimo da smo najveći broj (41) »izgurali« na kraj tabele gdje ima svoj konačni položaj. Zbog toga u slijedećim prolazima kroz tabelu nećemo više ispitivati zadnji element tabele. Nakon drugog prolaza izgled tabele je:

```
12  9  17  22  41
```

Sl. 5.

Vidimo da je slijedeći najveći broj već bio na svom mjestu, ali kako to računalo može »vidjeti«? Jedino tako, da izvrši sva ispitivanja i prebacivanja, bez obzira na nama ranije poznati položaj pojedinog broja. Nakon drugog prolaza kroz tabelu možemo »zaboraviti« na zadnja 2 člana koji su na svom konačnom položaju. U trećem prolazu ispitujemo samo prva 3 člana pa je nakon 3 prolaza izgled tabele:

```
9  12  17  22  41
```

Sl. 6

Iako je tabela sada u svom konačnom redoslijedu, računalo to ne može »vidjeti« i mora još međusobno usporediti prva 2 člana. Do izmjene mjesta neće doći jer su brojevi u rastućem redoslijedu. Evo kako izgleda ovakvo razvrstavanje napisano u BASIC-u za Spectrum:

```
 10 DIM T (50)
 20 LET  U=50
 25 FOR N=1 TO U
 27 LET T(N)=INT (RND *50)
 28 NEXT N
 60 FOR K=U TO 2 STEP – 1
 70 FOR N=1 TO K–1
 80 IF T(N) > T(N+1) THEN GOTO 500
 90 NEXT N
100 NEXT K
150 STOP
500 LET C=T (N+1)
510 LET T(N+1)=T(N)
520 LET T(N)=C
530 GOTO 90
```

Sl. 7



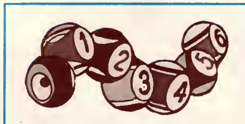
Radimo s tabelom od 50 elemenata. Punimo je slučajnim brojevima koje dobijemo RND (random) funkcijom BASIC-a. Ovako dobiveni brojevi nalaze se u rasponu od 0 do 49, zbog karakteristike INT funkcije koja zaokružuje broj na prvi cijeli broj koji je manji ili jednak broju argumenta. Proces uspoređivanja se vrši naredbama 60 do 100. Imamo, za tabelu od N elemenata, N-1 prolaz kroz tabelu.

Uvijek uspoređujemo 2 susjedna elementa, pa ako je prvi element veći od slijedećeg, vrši se izmjena mjesta (tih dvaju elemenata) u linijama 500 do 530. Kod izmjena mjesta potrebno nam je pomoćno polje, koje smo nazvali C.

Karakteristika ovoga razvrstavanja je da je vrlo jednostavno za programiranje i ne zauzima mnogo mjesta u memoriji. Slaba točka je da vrši veliki broj ispitivanja i prebacivanja u memoriji, što utječe na duljinu izvođenja razvrstavanja. Ujedno vrijeme izvođenja raste približno s kvadratom broja podataka koje razvrstava (tabela 1.) Mjerenje vremena smo vršili pomoću funkcije f = (65536 PEEK 23674+256 PEEK 23673 + PEEK 23672) / 50 (Vidi knjigu ZX SPECTRUM BASIC). Ovako dobiveno vrijeme je u sekundama. Ako uzmemo da je za razvrstavanje 100 podataka prosječno vrijeme 137 sekundi, za 200 podataka mjereno vrijeme 570 sekundi, onda bismo ekstrapolacijom za 400 podataka dobili vrijeme do 2280 sek. ili 38 minuta, dok bismo za 800 podataka trebali već više od 2 sata (točno 2 sata i 32 minute).

Ne mijenjajući osnovnu metodu razvrstavanja, samo izbjegavanjem stalnog premještanja podataka u memoriji, možemo znatno ubrzati proces. Pogledajmo slijedeći program:

```
 10 DIM T(50): LET U = 50
 20 FOR N = 1 TO U
 25 LET T(N) = INT (RND *U)
 30 NEXT N
 60 FOR K = U TO 2 STEP-1
 65 LET A = 1:LET C = T(A)
 70 FOR N = 1 TO K
 80 IF C < = T(N) THEN GOTO 90
 85 LET C = T(N):LET A = N
 90 NEXT N
 91 LET T(A) = T (K)
 93 LET T(K) = C
100 NEXT K
```



Što smo ovdje promijenili u odnosu na prvi primjer? Uveli smo novu varijablu A, koja je u stvari broj elemenata s najvećom vrijednosti u usporedbi, unutar jednog prolaza tabele. Za razliku od prve verzije programa, izmjena mjesta s elementima tabele se ne vrši, osim na kraju svakog prolaza (linije 91,93). Umjesto izmjene mijenja se polje C i polje A. Polje C uvijek ima vrijednost trenutno najvećeg elementa usporedbe, a polje A ima vrijednost položaja tog broja u tabeli. Osim što manji broj izmjena mjesta ubrzava rad, uvođenje polja C (umjesto da pišemo T(A) ubrzava rad jer programski tumač ima manje posla s običnim poljem nego s poljem tabele. Vrijeme razvrstavanja znatno je manje nego prvo, što ukazuje na velike mogućnosti da se korištenjem raznih programskih rješenja optimalizira vrijeme izvođenja. Ova metoda se može dalje razrađivati uvođenjem novih pointera ili efikasnijim pisanjem procedure, ali više se ne može bitno utjecati na vrijeme izvođenja razvrstavanja. Vidimo iz tabele 1 (razvrstavanje 2) da vrijeme razvrstavanja također raste s kvadratom broja podataka za nj.

3. RAZVRSTAVANJE SA STVARANJEM UREĐENIH PAROVA ELEMENATA TABELE

Ovu metodu prikazujemo zbog interesantne ideje razvrstavanja podataka. Prilagođeni oblici ovakvog načina razvrstavanja mogu se naći u sort programima na većim računalima. Ideja je da nerazvrstane podatke tretiramo kao grupe od po 2 elementa. Svaku grupu sortiramo. Nakon toga po dvije susjedne grupe od po 2 elementa spojimo u jednu od 4 razvrstana elementa. Tako povežemo sve elemente tabele. Spajanje grupa vršimo tako dugo, dok tabela nije podijeljena u samo dvije razvrstane grupe. Nakon toga ide konačni prolaz koji stvara jednu jedinu razvrstanu grupu, našu tabelu.

NEUREĐENA TABELA	17 12 41 9 22 16 3 11
1 prolaz	12 17 9 41 16 22 3 11
2 prolaz	9 12 17 41 3 11 16 22
3 prolaz	3 9 11 12 16 17 22 41

Sl. 9 Razvrstavanje sa stvaranjem uređenih parova

LISTA SORT PROGRAMA

```
10 DIM T(50) : LET U = 50
50 LET A = 1
60 LET M = INT (LN(U)/LN(2) +1)
70 FOR J = 1 TO M
75 LET B = A : LET A = 2*A
80 FOR I = 1 TO U-B STEP A
85 LET S = I + B : LET V = I + A-1 : IF V > U
THEN LET V = U
 90 FOR L = I TO S-1
 95 IF T(L) >T(S) THEN GO TO 500
100 NEXT L
110 NEXT I
120 NEXT J
150 STOP
500 LET C = T(S)
505 FOR K = S TO L + 1 STEP -1
510 LET T(k) = T (k-1)
515 NEXT k
520 LET T(L) = C
525 LET C = T(S)
530 FOR K = S + 1 TO V
540 IF C< = T(k) THEN LET T(k-1) = C : GO TO
100
550 LET T (k-1) = T(K)
560 NEXT K
570 LET T(V) = C:GO TO 100
```

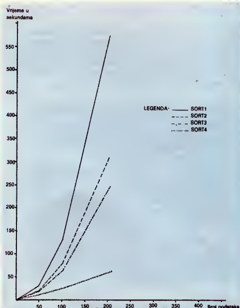
Sl. 9a Razvrstavanje sa stvaranjem sortnih lanaca

Ova metoda značajno smanjuje broj ispitivanja i prebacivanja podataka jer se u svakom slijedećem prolazu kroz tabelu uzima u obzir uređenost podataka iz prethodnog prolaza.

Kod ovakvog načina sortiranja pojavljuje se potreba za pomicanjem grupa elemenata tabele pri ubacivanju novog elementa u već razvrstanu grupu (linije 500 do 570). To, i nešto veća složenost algoritma, umanjuje prednost smanjenog broja ispitivanja. Vremena potrebna za sortiranje podataka prikazana su u tabeli 1 pod kolonom sort 3.

Broj prolaza kroz cijelu tabelu definiran je na slijedeći način. Za N elemenata tabele traži se najmanji K za koji vrijedi $N < 2^K$. Logaritmiranjem ovog izraza dobije se za K LnN/Ln2.

U liniji 60 umjesto K imamo M i koristimo INT funkciju BASIC-a. Za 50 elemenata tabele, za M dobijemo vrijednost 6.



4. »SHELL SORT«

Na kraju ćemo opisati još jedan način razvrstavanja koji zbog svoje jednostavnosti i efikasnosti zaslužuje posebnu pažnju. Ideja je da se međusobno ne ispituju 2 susjedna podatka, već 2 podatka koji su udaljeni za raspon K.
Svi elementi koji su međusobno udaljeni za interval K tvore podtabelu koja se sortira na način što odgovara prilagođenoj verziji sorta 1 (sl. 10).
Na slici 11. tabelu od 8 elemenata podijelili smo s intervalom K = 5 u tri tabele od 2 člana (1. i 6, 2. i 7. te 3. i 8. element), i 2 elementa koji u ovom prolazu nemaju svog para.

```
 10 DIM (50)
 15 REM TABELA SE PUNI SLUČAJNIM BROJEVI-
MA
 •
 •
 •
 50 FOR K = 1 TO N-1
 60 FOR I = K TO 1 STEP-1
 70 IF T(I) < = T(I + 1) THEN GO TO 90
 80     LET     C = T(I):LET     T(I) = T(I + 1):LET
T(I + 1) = C
 90 NEXT I
100 NEXT K
120 STOP
```

Sl. 10 Prilagođena verzija jednostavnog sorta (sl. 7)

Kad smo uspoređivanjem prešli cijelu tabelu, raspon K se smanjuje na 3. Ponovno se prođe cijela tabela, raspon K se ponovno smanji (ovaj put na 1) i tako se dođe do konačnog raspona K = 1.

Tabela ima 8 elemenata i za početni raspon odaberimo 5.

Početno stanje		Prvi prolaz		Drugi prolaz		Razvrstani podaci
17		16		9	Raspon	3
12	Raspon	3	Raspon	3	K = 1	9
41	K = 5	11	K = 3	11		11
9		9		12		12
22		22		22		16
16		17		17		17
3		12		16		22
11		41		41		41

Određivanje početnog intervala značajno je za vrijeme trajanja sorta. U literaturi se za početni interval najčešće upotrebljava izraz:

Sl. 11 »SHELL SORT«

$$I = 2^{INT\ (LOG(N)\ /\ LOG\ (2)}\ /\ -1$$

Ovako se osigurava da su početni i svi slijedeći intervali neparni brojevi. Svaki slijedeći interval se određuje sa I = INT (I/2). Prema tome za tabelu od 50 elemenata početni interval je 31, $(I = 2^5 - 1)$.
Intervali koji slijede su 15, 7, 3 1.
Iz priloženih programa vidimo da je razlika između jednostavnog sorta (Sl. 10) i SHELL sorta (Sl. 12) samo u 2 linije (20 110). .
Interval je u varijabli D.

```
 10 DIM T(50)
 15 REM TABELA SE PUNI SLUČAJNIM BRO-
JEVIMA
 20 LET D = 2↑ INT (LN (N) / LN (2)) -1
 50 FOR K = 1 TO N-D
 60 FOR I = K TO 1 STEP-D
 70 IF T (I) < = (I + D) THEN GO TO 90
 80 LET C = T(I):LET T(I) = T(I + D) : LET T
(I + D) = C
 90 NEXT 1
100 NEXT K
110 LET D = INT (D/2) : IF D > 0 THEN GO
TO 50
120 STOP
```

Sl. 12 »SHELL SORT«

Mjerena vremena razvrstavanja daju za ovu metodu daleko bolje rezultate u odnosu na druge ispitivane metode. Vidi se da rast vremena sortiranja povećanjem broja podataka nije proporcionalan sa kvadratom broja elemenata tabele. Ispitivanja pokazuju da porast vremena odgovara odnosu $N^{1,\ 26}$, gdje je N broj elemenata tabele. Možemo zaključiti da ova metoda sortiranja daje odlične vremenske rezultate, a ujedno je i vrlo jednostavna za programiranje.

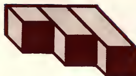
5. ZAKLJUČAK

Ispitali smo 3 različite vrste razvrstavanja. Na jednoj smo vrsti pokazali da se optimalizacijom iste metode može značajno smanjiti vrijeme sortiranja. Sve metode koriste vrlo malo memorije izvan tabele, tako da su pogodne za računala s malo memorije. Za broj podataka veći od 400 praktičnu primjenu ima jedino »shell sort« (na računalu snage ZX Spectrum) jer vrijeme sortiranja postaje ograničavajući faktor obrade.

Eduard Kovačić



SPECTRUMOVE TAJNE: EKRAN



Ekran je izuzetno važna jedinica malih računala preko koje ona najlakše predaju korisniku podatke iz svoje memorije. Podaci koji se kod mikroračunala ispisuju na ekranu (najčešće televizijskom), moraju se čuvati u posebnom dijelu memorije tzv. pretincu ekrana (Display File). Svake pedesetinke sekunde (evropski standard), kad televizor počinje iscrtavati novu sliku, podaci se iz pretinca šalju televizoru kao slijed njemu razumljivih signala. Pretinac ekrana može biti organiziran po znakovima, tako da svaki bajt predstavlja jedan znak i tada se na ekranu može ispisivati tekst i koristiti grafika niske raščlanjenosti (resolution) ili na drugi način koji omogućava korištenje grafike visoke raščlanjenosti, a svaka je točkica na ekranu predstavljena s 1, 2, 3 i 4 bita u memoriji, ovisno o tome koliko se boja na ekranu želi dobiti. U ovom tekstu pozabavit ćemo se ovim drugim načinom, konkretno Spectrumovom organizacijom ekrana. Spectrum ima gustoću 192 × 256 točaka koje se mogu pojaviti u osam boja, dvije nijanse osvijetljenosti i po želji mogu treperiti (»blinkati«). Slika se sastoji od točno 49152 točke, a budući da je za osam boja potrebno tri bita po točki ($8 = 2^3$), proizlazi da za čitav ekran Spectruma treba 147.456 bitova ili 18 K memorije. Kad bismo tome pribrojili bitove za osvijetljenost i treperenje, dobili bismo još veću brojku. Međutim, u stvarnosti Spectrum za ekran troši samo 6,75 K. Kako je to moguće?

»Štos« je u slijedećem: Spectrum u memoriji čuva samo podatke da li je pojedina točka u boji »tinte« (INK) ili pozadine (PAPER), što zahtijeva samo 6K memorije. Preostalih 0,75K, odnosno 768 bajtova sadrži informacije o boji tinte i pozadine, osvijetljenosti i treperenju, za skupine od 8 × 8 točaka (64 točke sačinjavaju jedan znak). Ovih 768 bajtova nazivaju se atributi. Jedan bajt atributa izgleda ovako:

f b p p p i i i

Bit »f« određuje da li će znak treperiti, a to možete kontrolirati BASIC-ovom naredbom FLASH. Bit »b« određuje nijansu osvijetljenosti, a kontrolira se naredbom BRIGHT. Tri bita »ppp« određuju boju pozadine u znaku (naredba PAPER je mijenja), a bitovi »iii« određuju kojom bojom će se ispitati znak (kontrolira se naredbom INK).

Upišite na svom Spectrumu program:

```
10 FOR I = 16384 TO 22527
20 POKE I,255
30 NEXT I
```

Otipkajte RUN i promatrajte ekran! Program stavlja broj 255 (koji označava da su sve točke u određenom bajtu u INK boji), redom od početka pretinca ekrana (16384) do početka atributa (22528) u memoriji, no ekran se puni na prvi pogled potpuno nelogično. No, ako bolje pogledate, vidjet ćete da određeno pravilo ipak postoji. Prvo se popunjava nulti red, zatim osmi, šesnaesti itd., sve do 64. Kad se popuni šezdeset i četvrti red, počinje se popunjavati prvi, deseti, sedamnaesti itd.

Kad se na taj način popuni čitav odsječak, na isti način se popune drugi i treći. Adresa nekog bajta na ekranu izgleda:

01 0 s s p p p	r r r c c c c c
viši bajt	niži bajt

Prva tri bita su uvijek 010. Dva bita označena sa »ss« određuju odsječak u kojem se nalazi traženi bajt. Tri bita »ppp« određuju red u skupini od 8 redova, odnosno red točkica u redu znakova (svaki znak se sastoji od 8 redova točkica). Tri bita označena s »rrr« određuju red znakova u odsječku. Zadnjih pet bitova »ccccc« određuju stupac u kojem se nalazi traženi bajt. Poznavajući sve podatke možemo napisati formulu koja iz koordinata točke izračunava njenu adresu na ekranu.

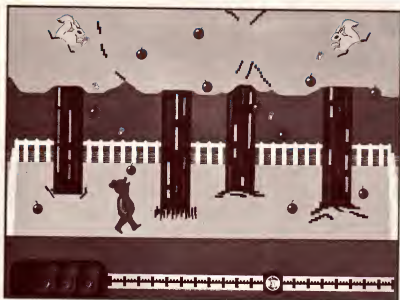
Ona glasi:

LET a = 16384 + 2048*INT (Y164) + 256*(Y-8*INT(y/8)) + 4*(8*INT(y/8) − 64*INT (y/64)) + INT(x/8)

Ako varijable x i y sadrže x i y koordinatu točke na ekranu, formula će u varijabli »a« dati adresu bajta u kojem se točka nalazi. Da saznamo koji bit u bajtu predstavlja traženu točku, upotrebljavamo formulu:

LET b = x − 8 * INT (x/8)

Varijabla »b« sadrži redni broj bita u bajtu (gledano s lijeva na desno), koji predstavlja zadanu točku. Možda će se sada pokoji čitalac upitati ima li smisla raditi s tako kompliciranim formulama, kada je Spectrum opskrbljen zaista moćnim naredbama za rad s grafikom, koje su između ostalog, bar pet puta brže od tih formula? Naravno, formule su spore i komplicirane, no prevedene u strojni jezik postaju znatno brže, pa čak i jednostavnije. S druge strane, Spectrumove naredbe ne omogućavaju ispis na bilo kojem mjestu na ekranu, niti omogućavaju crtanje u donja dva reda i još mnogo drugih sitnica. Stoga je mnogo zgodnije imati vlastite rutine za rad s ekranom, a ako želite praviti akcione igre, to postaje prijeka potreba.



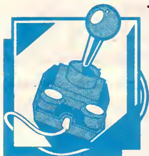
Rutina koja iz koordinata izračunava adresu, može u strojnom jeziku izgledati ovako:

```
LD      A,B
AND     A
RRA
SCF
RRA
AND     A
RRA
XOR     B
AND     #F8
XOR     B
LD      H,A
LD      A,C
RLCA
RLCA
RLCA
XOR     B
AND     #C7
XOR     B
RLCA
RLCA
LD      L,A
LD      A,C
AND     #07
RET
```

Ova rutina se razlikuje od Spectrumove rutine samo po tome što omogućuje crtanje i u donja dva reda, a kao ishodište odnosno točku s koordinatama 0,0, uzimaju gornji lijevi kut ekrana, dok Spectrum uzima donji lijevi. Prije pozivanja rutine treba u B registar staviti y, a u C – x koordinatu točke. Rutina vraća u HL paru registara, adresu bajta, a u A registru broj bita u bajtu koji predstavlja točku. Rutina je vrlo bitna i može se koristiti u igrama koje zahtijevaju veliku brzinu. Naravno, da biste mogli unijeti rutinu u računalo, poželjno je da imate neki strojni jezik, npr. GENS ili ZEUS.

Navedena rutina sama za sebe nema neku praktičnu upotrebu. No, u suradnji s drugim rutinama može vam omogućiti da maksimalno iskoristite Spectrumov, naizgled vrlo nezgodan, ekran.

IGOR FISCHER



Palice za igru i njihovi priključci

Jedan od prvih perifernih uređaja, koje ste poželjeli nabaviti za vaše računalo, vjerojatno su palice za igru *(joysticks)*. Nakon što ste sate i sate proveli uz novu igru, za koju su vam bile potrebne barem tri ruke, ustanovili ste da je jedini spas vašoj, od silnog udaranja, ugroženoj tastaturi kupovina barem jedne »prave« palice za igru. To bi akcionim, pa i drugim, igrama pridodalo novu dimenziju i veći osjećaj realnosti, a vas oslobodilo zamornog traženja odgovarajuće tipke po tastaturi.

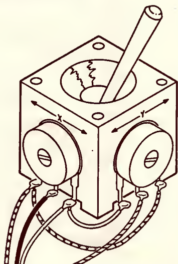
Ako ste vlasnik Atarijevih ili Commodorovih računala, tada je jedini problem koju palicu odabrati od mnogih koje se nude. Međutim za Spectrum, BBC, Electron i neka druga računala problem je pronaći palicu koja će zaista i raditi. Iako su palice dostigle najfantastičnije izvedbe *(vizualno i sklopovski)* počet ćemo od onih najjednostavnijih.

Palice za igru sastoje se od baznog kućišta i ručice, čije pokretanje kontrolira pomicanje određenog grafičkog objekta na zaslonu TV aparata ili monitora. Obzirom na način kako se mehaničko pokretanje ručice pretvara u informaciju, koja se predaje računalu, palice u osnovi dijelimo na digitalne i analogne uređaje. Kod digitalnih palica priključna veza je standardizirana (Atarijev standard), tzv. »D« priključak s 9 nožica (5 u jednom i 4 u drugom redu). Pomak ručice ili pritisak na tipku za okidanje, aktivira određeni mikroprekidač, koji kratko spaja zajedničku liniju i liniju koja odgovara pomaku. Na taj način se na sabirnicu podataka *(data bus)* predaje digitalni kôd izvršene akcije. Ukoliko je pojedini bit 0, znači da je bio uključen pripadajući mikroprekidač. Kod takvih palica veoma je važno da vrijeme odgovara *(response time)*, odnosno reagiranje bude što kraće i da se ručica, nakon što se otpusti, automatski vrati u početni položaj.

Analogna palica na izlazu daje analognu informaciju – napon određene veličine koji se kontinuirano kreće u rasponu od minimalne (OV) do neke maksimalne vrijednosti. Pokretanje ručice prijenosnim sistemom poluga zakreće dva promjenljiva otpornika (potenciometra) i time mijenja njihovu vrijednost. Svaki od potenciometara kontrolira jednu koordinatu dvodimenzionalnog prostora. Potenciometri su tako povezani da je u trenutku kad je palica u nekom od krajnjih položaja (lijevo, desno, naprijed ili natrag) jedan od njih u stanju maksimalnog otklona, dok je drugi u početnom položaju. Svaki se položaj ručice prevede u X i Y koordinatu s odgovarajućom vrijednošću otpora. Na taj način može se registrirati položaj ručice u bilo kojem smjeru. Postoje i uređaji s jednim potenciometrom *(paddles)* koji umjesto ručice imaju zakretni valjak priključen direktno na osovinu potenciometra, te omogućuju pokretanje samo u dva smjera (lijevo, desno). Kako su računala digitalne naprave, analogni signal potrebno je pretvoriti u digitalni. U tu svrhu koriste se analogno-digitalni (A/D) pretvarači, koji analognu veličinu prevedu u višebitnu digitalnu informaciju. Od popularnijih kućnih računala jedino BBC ima ugrađen A/D pretvarač, te se samo na njega mogu direktno priključiti palice za igru analogne izvedbe.

DIGITALNE PALICE

Od digitalnih palica svakako su najpoznatije Atarijeve, koje su svojom niskom cijenom ($6,95) zavladale tržištem. Međutim, njihova naizgled robusnost može zavarati igrača da primijeni prevelik pritisak i slomi ručicu. Tipkalo za pucanje smješteno je isuviše lijevo na kućištu što ne pogoduje ljevorukim igračima. Unatoč svemu, nabavka ovih palica, obzirom na njihovu nisku cijenu koja ulazi u naše carinske okvire, predstavlja dobar posao.



1. ATARI – pomična kugla
2. KRAFT – vrlo osjetljiva palica
3. STARFIGHTER – čvrsti model
4. ATARI – jeftina palica
5. JOY-SENSOR – reagira na dodir



6. LE STICK – nedovoljno osjetljiva
7. SINCLAIR – međusklop (interface)
8. DOWNSWAY – međusklop koji se može programirati



Starfighter palica je manja, ali i manje osjetljiva od Atarijevog modela, pa se može preporučiti igračima snažnije ruke ili djeci. Cijena im je, međutim, dvostruko veća ($ 13,95). Vrlo su dobre **Kraft** palice za igru. Ručica im je tanja nego što je uobičajeno, ali vrijeme reagiranja je trenutačno. Nakon otpuštanja, ručica se sama vraća u središnji položaj. Kraft palice su opremljene dugom priključnom žicom, gotovo 3 m u odnosu na 1,5 m koliko standardno imaju drugi modeli. Cijena im je također 13,95$.

Slijedeća dva modela reklamiraju se kao »najbolje prodavane palice na američkom tržištu«. To su **Triga Command,** koji se prodaje zajedno s međusklopom (*interface*) za Spectrum, o čemu će kasnije biti više riječi, za $19,99 i **Super Champ.** Karakteristika ovih palica je da imaju tipke za pucanje na vrhu ručice, a Super Champ ima još jednu ispod samog vrha. Super Champ ima okruglo kućište u koje se automatski uvlači višak priključnog voda. Vrlo je robustan, a cijena mu je $12,95.

Poznata firma **Kempston** nudi tri modela PRO1000 ($10,50), PRO3000 ($12,75) i PRO5000 ($13,50), od kojih ovaj posljednji ima ručicu što podsjeća na mjerač brzina u automobilu. Modeli ovog proizvođača poznati su po kvaliteti, jedino je hod ručice prekratak.

I japanski proizvođači kućnih MSX računala, koji su standardno opremljeni Atari spojnim priključnicama za palice, nude čitav niz palica koje se ističu privlačnim izgledom kao što su **Sanyo MJY002** ($12,95), **JVC HCJ615** ($12,95), **Toshiba HX-J400** ($12,95) i posebno **Sony JS55** ($19,95).

U oblikovanju, pa i kvaliteti, najdalje je otišla Quickshot serija palica za igru. Mnogi proizvođači međusklopova za računala uzimaju ih kao standardne i nude ih u svojim proizvodnim paketima. Opremljene su vakuumskim držačima na donjoj strani kućišta, čime je jedna ruka oslobođena za eventualne akcije na tastaturi. **Quickshot I** ($9,95) posjeduje čak tri tipke za pucanje, dok ekskluzivniji model **Quickshot II** ($10,95) nudi tipke za stalno pucanje (*rapid fire*), pojedinačno okidanje (*trigger*) i automatsku vatru. U najavi je i **Quickshot III** s dvostrukim skupom tipki i posebno ugrađenom numeričkom tastaturom.

ANALOGNE PALICE

Od analognih kvalitetne su palice **Harrier** ($10,95), koje po svojoj konstrukciji podsjećaju na Quickshot I model. Ekskluzivni model **Delta 14b** ($14,95) osim ručice ima i 14 tipki koje korisnik može programirati.

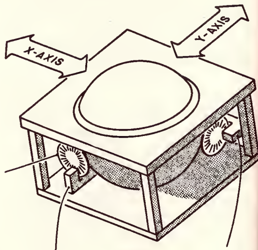
PALICE ZA IGRU

Sve o palicama – od jednostavnih do eskluzivnih modela. Kako izgledaju, koliko koštaju, kako ih nabaviti, koliko vrijede. Uporedni test analognih i digitalnih palica. Prikaz najnovijih uređaja – pomične kugle i infracrvenih predajnika. Savjet vlasnicima najpopularnijih kućnih računala – koja palica za koje računalo.

Međutim, da bi se iskoristile sve mogućnosti ovog modela potrebno je, ukoliko ga želimo priključiti na BBC računalo, nabaviti još dodatni međusklop ($14,85) i program ($5,95) što ga čini preskupim. I za Commodorove modele računala može se nabaviti analogna palica s ugrađenim A/D pretvaračem **VS64-0902** za $14,95.

Osim klasičnih palica za igru u novije se vrijeme pojavljuju sve moderniji uređaji za kontrolu video igara. Međutim, kod tih je uređaja problem što su usko specijalizirani za pojedini tip igre iako se to u reklami ne može pročitati. Jedan od takvih proizvoda je i Le **Stick** koji liči na standardnu palicu, ali nema bazno kućište, tako da nad njim imate potpunu trodimenzionalnu kontrolu dok vam je druga ruka slobodna. Izveden je pomoću posebnih živinih mikroprekidača koji su osjetljivi na pomak (tilt) tako da ga jednostavno nagnete u pravcu kojim želite ići. Idealan je za klasičnu Atarijevu igru Space Raiders, dok je u višim razinama te igre premalo osjetljiv. Cijena mu je također ekskluzivna $24,95.

Interesantan uređaj je i Atarijev Track-Ball. Sastoji se od pomične kugle, smještene u kućištu, koju možete dlanom okretati na sve strane. Unutar kućišta nalaze se dva svjetlosna uređaja, po jedan za svaku koordinatu, koji se sastoje od izvora svjetlosti, perforiranog rotirajućeg diska i fotoćelije. Bilo koji pomak kugle uzrokuje pomak barem jednog diska, koji naizmjenično prekida svjetlosni snop, te se na taj način registrira pomak duž X i Y osi od početnog položaja. Međutim, ovaj uređaj je od koristi jedino u igrama Centipede i Missile Command, dok mu je cijena dosta visoka ($19,99).



Toj grupi pripadaju i uređaji koji za pomak u željenom smjeru imaju na pritisak osjetljiva tipkala (*touch-sensitive*). Takvi su **Joy Sensor ($29,95)** i RAT ($29,95 zajedno s međusklopom za Spectrum). Međutim, njihova upotrebna vrijednost je diskutabilna jer ćete gotovo redovito dodirnuti pogrešnu tipku. RAT uređaja ima ugrađen infracrveni predajnik, tako da je to u stvari daljinski upravljač (Remote Action Transmitter), dok je prijemnik ugrađen u međusklop. Proizvođač garantira djelotvornost uređaja i na udaljenosti većoj od 10 m od prijemnika, što je zapravo od male vrijednosti kad se zna da se većina akcionih igrača provede na udaljenosti manjoj od pola metra od ekrana.

Nakon što ste, obzirom na cijenu i kvalitetu, odabrali onu »pravu« palicu, treba prije nego što je doista i kupite, dobro provjeriti da li je uopće možete priključiti na računalo. Ako je vaše računalo Commodore, Atari, neki od MSX ljepotana ili bilo koje drugo koje posjeduje standardnu Atarijevu priključnicu, tada će svaka od navedenih digitalnih palica odgovarati. Vlasnici BBC računala treba da biraju između analognih palica ili mogu odabrati neku od digitalnih koje se mogu priključiti jedino na korisnički ulaz (*user port*), naravno, ako posjeduju odgovarajući priključak (takve nismo opisali). Međutim, vlasnici Spectruma, a takvih je u našoj zemlji najviše, ako žele nabaviti palicu, morat će nabaviti i poseban međusklop (*interface*) koji će omogućiti povezivanje palice i njihovog ljubimca (ali o tome više u idućem broju).

Tomislav Žganec



CIJENE PALICA I ADRESE PROIZVOĐAČA

Atari Joystick	$ 6,95	Atari International (U.K.) Ltd, Atari House, Railway Terrace, Slough, Berkshire
Atari Super Controller	$ 14,95	
Trak Ball	$ 19,95	
Starfighter Joystick	$ 13,95	Consumer Electronics Ltd, Failsworth, Manchester M35 OHS
Kraft Joystick	$ 13,95	
Joy-Sensor	$ 29,95	
Triga Command + međusklop	$ 19,99	Datel Electronics, Unit 8, Fenton Industrial Estate, Dewsbury, Road, Fenton, Stoke-on-Trent
Super Champ	$ 12,95	Dean Electronics, Limited, Gilendale Park, Fernbank, Road, Ascot, Berks
PRO 3000	$ 12,75	Kempston Micro Electronics Ltd, Singer Way, Woburn Road Industrial Estate, Kempston, Bedford MK 42 7AF
PRO 5000	$ 13,50	
MJY002	$ 12,95	Sanyo
HCJ615	$ 12,95	JUC
HX-J400	$ 12,95	Toshiba
JS55	$ 19,95	Sony
Quickshot I	$ 9,95	CK Supples, Unit 5, Norside, Oldmixon Crescent, Weston-super-Mare, Avon, U.K.
Quickshot II	$ 10,95	
Redball Joystick	$ 27,95	Vicsoft, 675 Ajax Avenue, Slough, SL1 4BG, Berks
VS64-0902	$ 14,95	
Paddle Controllers (2)	$ 13,95	
Hamer	$ 10,95	Tradeink UK, Dept. MU21 Broughton St. Edinburgh EH1 3JU
Delta 14b	$ 14,95	Voltmace Ltd, Park Drive, Baldock, Herts, SG7 GEV
Le Stick	$ 24,95	Maplin Electronics, PO BOX 3, Ray leigh, Essex SS6 8LR
RAT + međusklop	$ 29,95	Cheetah Marketing Ltd. (Dept. 2x), 24 Ray Street, London EC1R 3DJ

SAVJETI ZA BRZI BASIC

Kompanija MICROSOFT, jedna od vodećih, ako ne i vodeća tvrtka na području izrade programa za mikroračunala, prodaje BASIC u različitim varijantama od kojih su najpoznatiji MBasic za korisnike upravljačkih sistema CP/M i BASIC-86 za 16-bitne procesore. Osim ovih, ista tvrtka proizvodi još desetak verzija programskih tumača (interpreter) za BASIC za više ili manje poznata računala. No bez obzira na kvalitete tumača, on je uvijek spor. Ovdje navodimo 10 savjeta kako bi se program ipak nešto ubrzao.

IZBJEGAVAJTE GO TO NAREDBE

Pokušajte izbaciti **GO TO** naredbu iz vašeg skupa (repertoara) naredbi, pogotovo ako se nalazi unutar neke petlje. Umjesto **GO TO** treba pisati FOR-NEXT i WHILE-WEND, jer na taj način ne gubimo vrijeme tražeći adresu linije navedene iza **GO TO** naredbe. Na taj način možemo uštedjeti i do 20% u izvođenju petlji, pogotovo ako se nalaze negdje na početku velikog programa.

Osim ove prednosti naredbe FOR-NEXT i WHILE-WEND su daleko preglednije i upotrebljavaju se za strukturirano programiranje čija glavna svrha je preglednost i razumljivost programa.

IZBJEGAVAJTE POTPROGRAME (PODRUTINE)

Iz istih razloga upotrebljavajte sistemske pozive (function calls) a ne podrutine gdje god je moguće.

U MBasicu korisnički definirane FN funkcije ograničene su na jednostavne izraze, ali se odgovarajućom spretnošću iz njih mogu izvući čuda. Rezultati su uvijek bolji od korištenja GOSUB-RETURN.

KORISTITE POLJA, A NE BROJEVE

Koristite nazive polja, a ne stvarne brojeve gdje god je moguće! Posebno se ovo pravilo odnosi na kontrolne varijable i upisnike (subscript) tabela. Ukoliko radimo sa poljima, instrukcija strojnog nivoa lako pronađe adresu polja, dok stvarne brojeve tumač mora raspakirati i dekodirati. Koristeći ovo pravilo pretraživanje tabele ubrzat će se do 25%.

SMANJITE BROJ LINIJA

Smanjite broj linija. Stavite u jednu liniju više naredbi. Ipak razmislite o ovoj i slijedećoj preporuci. U ova dva slučaja cijena ubrzanja programa je smanjenje preglednosti i čitljivosti programa.

ČUVAJTE PROGRAME U BINARNOM OBLIKU

Pri spremanju programa upotrebljavajte binarni, a ne ASCII zapis što znači da ne koristite A mogućnost (opciju) uz **SAVE** naredbu. Premda se brzina izvođenja ne poboljšava, skraćuje se punjenje programa u memoriju i njihovo međusobno povezivanje i do tri puta.

NAPIŠITE ŠTO KRAĆI PROGRAM

Smanjite broj izvornih naredbi izbacujući komentare i praznine. Skratite nazive polja. Svaki put tumač pretražuje i obrađuje svaki znak na koji naiđe kad se izvodi naredba.

IZBACITE POGREŠKE U TOČKAMA KOJE NISU KRITIČNE

Programski tumač u normalnoj situaciji prepravlja (reorganizira) svoj znakovni niz prazinina samo kad je potrebno i to vrlo polako. Naredbom poput

DUMMY = PRE ("")

možemo izazvati reorganizaciju. Ova naredba neće odstraniti potrebu za izbacivanjem pogrešaka, ali ovako možete odrediti trenutak kad će se zbiti izbacivanje.

UPOTREBLJAVAJTE SLOGOVE DUGAČKE 128 ZNAKOVA

Pri korištenju datoteka s direktnim pristupom, koristite slogove dužine 128 znakova ili umnožak nekog broja i 128. **GET** sloga od 129 znakova (bytes) trajat će otprilike dvostruko od čitanja sloga dugačkog 128 znakova.

USAVRŠAVAJTE TEHNIKU PROGRAMIRANJA

Ukoliko u programu imate sortiranje, pretraživanje tabela, rad s indeksima ili spremnicima (buffers) treba u literaturi potražiti najefikasniji način za određeni problem. Ova znanja su općenita i ne primjenjuju se samo u BASIC-u.

KORISTITE PROGRAMSKI PREVODILAC (COMPILER) UMJESTO PROGRAMSKOG TUMAČA (INTERPRETER)

Ne koristite tumač za aplikacije koje su vremenski točno određene, niti za programe koji se često upotrebljavaju. Uz male ili nikakve promjene, programi koji se izvode pod kontrolom tumača mogu se izvoditi pod kontrolom odgovarajućeg prevodioca. Samo prevođenje je vrlo sporo, ali je zato izvođenje mnogo brže. Ostalih devet savjeta ubrzavaju izvođenje, ali u usporedbi s upotrebom prevodioca ubrzanja su gotovo zanemarljiva.

Savjet kao savjet može biti dobar ili loš, ali uvijek ostaje savjet. Praksa potvrđuje ili opovrgava. Zbog toga pogledajte programe i rezultate dobivene na Spectrumu. U ovoj reviji možete pronaći i napis o efikasnosti sortiranja.

Kao što ni u životu ništa nije bez razloga, tako je vrlo često efikasnost programiranja umješnost programera da ocijeni što ide na štetu čega a u čiju korist. Jednostavnije rečeno: da li povećati brzinu izvođenja čija je posljedica smanjenje razumljivosti ili čitljivosti programa? ili: isplati li se povećati brzinu izvođenja povećanjem potrebne memorije? Praksa i iskustvo najefikasnije pronalaze najoptimalnije odnose između pojedinih kategorija, jer faktori koji utječu specifični su u svakoj situaciji.

Pripremio: Željko Pažur

Računala mogu vidjeti



Digitalizacija i računalska obrada video slike je stvar relativno starijeg datuma, međutim cijena jednog takvog uređaja nije dopuštala široku upotrebu i primjenu u sistemu s mikroračunalom. Većina uređaja za računalsko snimanje i obradu slike koristi običnu crnobijelu TV kameru kao optički senzor. Analogni signal na video izlazu kamere mora se prevesti u digitalni signal razumljiv sklopovlju računala. Kako se svake sekunde pretraži 25 kompletnih okvira (frames) slike, dobiva se analogni signal visoke frekvencije (5 MHz) koji se zatim vrlo brzim A/D (analogno digitalnim) pretvaračima digitalizira i sprema u brzu poluvodičku memoriju. Na taj način se mogu obraditi slike vrlo visoke raščlanjenosti koje se naglo mijenjaju u vremenu. Međutim, cijena takvog uređaja prelazi $ 10.000.

Manje inteligentni sistemi, a time naravno i jeftiniji, koriste se ukoliko promatrani objekt ostaje nepomičan dovoljno dugo da se slika može obraditi jednostavnijim i sporijim uređajima (0.1-1 MHz). Takve sisteme ste vjerojatno i vidjeli, a koriste se između ostalog i za štampanje računalskih plakata i slika na majicama. Cijena im je oko $ 1.000, što je još uvijek mnogo.

Problem s dosad navedenim sistemima je taj što su se analogni uređaji upotrebljavali u digitalnim aplikacijama. Mnogo povoljnije bi bilo upotrijebiti video uređaj koji bi radio na digitalnom principu s poluvodičkim elementima čiji se izlaz mijenja kao funkcija intenziteta svjetlosti (1 ili 0). Na taj način rade novi računalski video sistemi koji svojom prihvatljivom cijenom postaju prave »oči« sve popularnijih mikroračunala. Najpoznatiji takvi uređaji su Micro-D-Cam američkog proizvođača Micromint ($295), koji se proizvodi za Apple IIE i IBM PC računala, i Snap EVI engleskog proizvođača Micro Robotics ($130), koji se može priključiti na BBC.

pozicije. Ukoliko je svjetlost dovoljne jačine, ćelija će poprimiti vrijednost logičke nule. Nakon vremena ekspozicije, svakoj ćeliji se može individualno pristupiti i ispitati njezin sadržaj. Na taj način, kompletna slika je digitalizirana u niz jedinica i nula, te se pomoću sklopova i posebnog programa računala može prikazati na zaslonu TV aparata ili monitora. Ukoliko se slika želi prebaciti u radnu memoriju, koristi se poseban program koji vrši sažimanje područja jednoličnog intenziteta. Jednostavne siluete će tako zauzeti samo 1K memorije, dok bi u protivnom treba-



Osnovu tih sistema predstavlja IS32 optički DRAM (Dynamic Random Access Memory) poluvodički element. Dinamička memorija pamti informaciju na parazitnoj kapacitivnosti MOS (metal-oxyd semiconductor) tranzistora. Zbog nesavršenosti, parazitna kapacitivnost se s vremenom izbija. Ukoliko memorijsku ćeliju izložimo svjetlosti, fotoni će ubrzati taj postupak. IS32 je memorijski element koji sadrži 65536 (64K) svjetlosno osjetljivih memorijskih ćelija, veličine 8×9 mikrona, koje su grupirane u dva dvodimenzionalna polja. Svako polje sadrži 32768 ćelija. To su dvije matrice sa 128 redova i 256 kolona koje odvaja linija široka otprilike 25 elemenata. Ta linija se odražava kao optički »mrtva zona«. Zbog jednostavnosti koristi se samo jedno polje, odnosno jedna matrica, ograničavajući pri tome raščlanjenost slike na spomenutih 256×128 elemenata (pixel). Silicijeva pločica, koja sadrži memorijske ćelije, nije u potpunosti zatvorena u keramičko kućište, već je s prednje strane izložena svjetlu preko kvarcnog poklopca.

Kamera, koja je sastavni dio sistema, usmjerava reflektiranu svjetlost s promatranog objekta kroz sistem leća na jedno od memorijskih polja. Sve ćelije su inicijalno nabijene (logička 1) i zbog djelovanja fotona dolazi do ubrzanog izbijanja kapacitivnosti, proporcionalnog s trajanjem ekslo čitavih 4K. Treba napomenuti da, kao i sve ostale dinamičke memorije, IS32 treba periodički osvježiti sadržaj (refresh). Za vrijeme osvježavanja, memorijske ćelije nisu osjetljive na svjetlost.

Kao što smo rekli, ovisno o jačini svjetlosti sadržaj memorijske ćelije će biti 1 do 0, što će rezultirati na slici u oštrim granicama između svjetla i sjene. Da bi se dobili sivi tonovi, objekt se snima 15 puta uzastopno i svaki puta je vrijeme ekspozicije nešto duže nego u prethodnom slučaju. Svaka ćelija ima pridružen brojač u kojem se vodi računa koliko puta je očitavanjem dotična ćelija imala vrijednost 1. Ukoliko je vrijednost brojača 15, odgovarajuća točka ekrana će biti maksimalno tamna, a u slučaju da je vrijednost 8, točka će biti u sivoj nijansi, na pola između crne i bijele. O tome se brine poseban program, koji je sastavni dio sistema.

Brzina kojom kamera može pretraživati pojedini objekt ovisi o intenzitetu svjetlosti. Ukoliko je intenzitet veći, kamera može brže raditi. Otprilike, brzina je oko 15 okvira u sekundi u slučaju maksimalne brzine.

Kada je jednom računalo prihvatilo sliku u memoriju, ono je može i analizirati. To je i bitna razlika između ovakvih uređaja i sistema video kamera/rekorder, koji daju bolju sliku, ali su potpuno neinteligentni. Sposobnost računala da obradi ono što je »vidjelo« otvara neslućene mogućnosti u raznim oblastima, počevši od zaštite, kontrole objekata u procesu proizvodnje, grafičkom oblikovanju, prepoznavanju oblika i znakova, robotici i na polju umjetne inteligencije. Možete li samo zamisliti robota koji osim što razgovora s vama za vrijeme partije šaha, vidi i pomiče figure po šahovskoj tabli?

Tomislav Žganec

Elektronični glazbeni instrumenti

Glazba i glazbena umjetnost jedno je od područja umjetnosti zahvaćenog naglim razvojem elektronike, što se ponajviše ogleda u širokom rasponu elektroničnih glazbenih instrumenata, koji se imenuju skupnim nazivom sintesajzeri (od engl. synthesizer). Sintesajzer je, kao što mu i samo ime kaže, sintetizator zvuka. Svi dosadašnji glazbeni instrumenti koriste se prirodnim odlikama materijala od kojih su napravljeni. To znači da materijal i način izrade određuju zvuk nekog glazbala. Zašto razlikujemo npr. klarinet od oboe ili trubu od trombona? Kažemo da ih razlikujemo, po specifičnosti boje zvuka. Specifična boja zvuka nekog instrumenta raspoznaje se zbog suzvuka, koji se pojavljuju uz osnovni ton. Da bismo to malo bolje pojasnili, potrebno je reći da niti jedan ton, koji izlazi iz nekog instrumenta nije potpuno (frekventno) čist. Najčišći ton u tom smislu ima flauta.

Uz taj osnovni ton pojavljuje se čitav niz tonova koji se nazivaju flažoleti ili harmonici. Broj i raspored tih suzvuka daje osobitost nekom instrumentu. Budući da se sve to može izračunati i frekventno izraziti, uz određeni izvor (generator) zvuka, sintezom osnovnog tona i dodavanjem suzvuka, može se kreirati (imitirati) zvuk, bilo kojeg glazbenog instrumenta.

Prvi sintesajzeri, koji su se pojavili prije petnaestak godina, bili su monofoni, tj., jednoglasni. Nemogućnost sviranja više od jednog tona istovremeno, bio je ograničavajući faktor, no, s vremenom, počeli su se pojavljivati sintesajzeri s više glasova, koji se mogu svirati istovremeno. Takve sintesajzere zovemo polifonim ili višeglasnim. Današnje tržište uglavnom nudi polifone sintesajzere, makar se može naći još pokoji monofoni.

Monofoni sintesajzeri koriste se najčešće za efektne solo-dionice, u kojima je prevladavajuća neka melodijska linija ili naglašena ritmička figura (Jan Hammer, George Duke, Tom Coster itd.). Mogućnost snimanja (različitih) sekvenci na više staza, a zatim stapanja na jednu, omogućuje da se kod polifonih sintesajzera broj glasova povećava po želji, što daje jednom izvođaču moć da zamijeni više glazbenika odjednom, pa i čitav orkestar. Ovdje se dotičemo jednog estetskog i sociološkog pitanja, budući da među glazbenicima, a i ne samo među njima, postoje velika razmimoilaženja u mišljenjima o svrsi i koristi sintesajzera. Da li on šteti ili koristi glazbi i izvođačima? O onima koji su protiv svega što je novo ne treba trošiti ni vremena, a ni prostora. Važniji su protivnici glazbenici, koji gube posao pred nastupom nove tehnologije. Budući da je jeftinije angažirati jednog glazbenika, nego više njih, producenti se prilikom snimanja ploča, TV i radio reklama, glazbe za filmove itd., nerijetko odlučuju za jeftinije i jednostavnije rješenje. Glazbenici, koji su uložili mnogo godina i truda, da bi ovladali nekim instrumentom, gube posao zbog sintesajzera. Pritom najčešće spominju grčkog klavijaturista Vangelisa, koji je napisao i izvodio glazbu u filmovima »Vatrene kočije« i »Blade Runner«. No, dok rasprave traju, proizvođači neumorno iz dana u dan, iznose na tržište sve više novih modela i uređaja, tako da je sve teže i teže snaći se u toj džungli ponuđenog.



Između mnoštva tvrtki, koje nude svoje proizvode, treba znati odabrati vrstu sintesajzera s obzirom na mogućnosti koje ima, spojivost s drugim instrumentima, da li se može spojiti na mikroračunalo itd.

MUZIKALNA RAČUNALA

...e analogni i digitalni sintesajzeri. U stvari, na ...u postoji svega nekoliko instrumenata koji su ...uno analogni. Većina ih je analogno/digitalni dok su neki u potpunosti digitalni. Ni jedna od dviju tehnologija nije sama po sebi superiorna ...goj. Svi instrumenti, koji se mogu programirati za ...ranjivanje podataka, koriste digitalnu memoriju. gotovo svim polifonim sintesajzerima klavijatura ...di s digitalnim krugovima (circuitry). Nakon te toč... e postoje razlike. Fizička oprema izvora zvuka ...ardware zvučnog generatora) može imati analog...e ili digitalne oscilatore, zatim digitalne oscilatore spojene s analognim filterima itd.



Danas većina sintesajzera daje izvođaču mogućnost programiranja, tj. stvaranja i spremanja u memoriju vlastitog zvuka, no ipak postoji još nekoliko instrumenata čiji su zvuci unaprijed tvornički programirani i ne mogu se mijenjati osim na neki jednostavan, neprogramski način.

Posebna kategorija sintesajzera je ona koja posjeduje digitalni uređaj za stvaranje uzoraka zvuka (digital sampling machine). Taj uređaj omogućuje potpuno vjernu rekonstrukciju zvuka akustičnih instrumenata jer sadrži digitalnu snimku stvarnih (realnih) zvukova. Neki uređaji omogućuju manipuliranje tim zvucima, koji onda postaju manje prepoznatljivi, a poprimaju više elektronično sazvučje.

Kod nekih polifonih sintesajzera, ali ne kod svih, postoji mogućnost dijeljenja klavijature na dva dijela, tako da je moguće programirati posebno lijevu, a posebno desnu stranu. Krajnji rezultat su dva potpuno različita zvuka s iste klavijature.



Važna stavka kod odabira sintesajzera je osjetljivost i brzina tipki klavijature. Kod jednostavnijih modela, tipka nije ništa drugo do običan prekidač. Klavijaturistima, naviknutim na instrumente (koji odmah po pritisku tipke daju zvuk) tipke (koje reagiraju na snagu pritiska prstima po klavijaturi) dakle, mogućnost dobivanja glasnijeg ili tišeg zvuka, nijansiranja tona, najčešće ne odgovaraju.

Takve klavijature su često i vrlo spore. No, postoji mnogo sintesajzera koji imaju vrlo profinjene klavijature, tako da mogu zadovoljiti i vrlo izbirljive glazbenike.

Postoje dva osnovna tipa osjetljivosti tipki. Osjetljivost na brzinu i osjetljivost na pritisak. Kod jednostavnijih elektroničnih instrumenata osjetljivost na brzinu znači samo jednu stvar, a to je da snažnijim udarcem po tipki ton postaje glasniji ili oštriji. Kod složenijih to može značiti mnogo više, ovisno o tome kakvi su sve podaci uprogramirani u osjetljivost na brzinu.

Kod nekih klavijatura tipke su opterećene na različite načine, da bi se dobio otpor na pritisak, kao kod klavira. Takve klavijature sadrže određen sustav sačinjen od raznih poluga i opruga ili čak pravih metalnih opterećenja da bi simuliranje otpora tipki kao kod klavira bilo što vjernije. Makar svi ti dodaci čine cijenu instrumenta većom, ako je izvođač već klavirist, može mu se takva klavijatura učiniti vrlo podesnom, jer mu omogućuje lakšu procjenu, koliko će brzina i pritisak prstiju po tipkama utjecati na ton. No, niti jedna klavijatura ne može u potpunosti simulirati otpor tipki kao kod klavira. Privikavanje na klavijaturu bez dodatnog opterećenja može zadovoljiti izvođača.

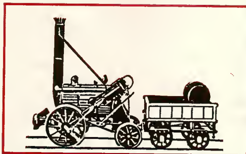
Jedno od najzanimljivijih područja spoja glazbe i tehnologije je povezivanje sintesajzera i mikroračunala te stvaranje glazbe računalskom tehnikom. Veza koja premošćuje udaljenst između mikroračunala i elektroničnih glazbenih instrumenata je **MIDI** protokol. **MIDI** je kratica za digitalni međusklop muzičkih instrumenata (Musical instrument digital interface). No, to je već tema za slijedeći broj.

Gordan Blečić

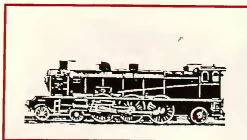
Ako želite programirati svoje računalo, morate se opredijeliti za neki od programskih jezika. Mnogi će se (unatoč svim nedostacima) odlučiti za BASIC zbog njegove jednostavnosti. Ako ste se već odlučili za BASIC, morate odlučiti hoćete li upotrebljavati tumač *(interpreter)* ili prevodilac *(compiler)*.

Programski tumač izvodi program »u živo« jer svaki put pri nailasku na neki podatak (naredba, literal, praznina ili bilo što drugo) prevodi ga u strojni kôd. Ako program prolazi kroz jednu petlju tisuću puta, tumač će svaki put prevoditi naredbe.

Programski prevodilac, naprotiv, prevodi program samo jednom, do prve izmjene. Nakon prevođenja



Programski tumač ili programski prevodilac



posebni program povezuje prevedene rutine programa sa sistemskim rutinama upravljačkog sustava. Tako objedinjeni program, u kojem svaka instrukcija i svako polje ima točno određenu adresu i prevedeni strojni kôd, spreman je za izvođenje. Samo izvođenje sastoji se od upućivanja na stalne adrese u memoriji i izvršavanja na tim mjestima računalu razumljivih zapisa (strojnog koda).

Razlozi za i protiv su lako mjerljivi. Prednosti tumača su brzo i jednostavno programiranje te jednostavno pronalaženje i ispravljanje pogrešaka. Tumač je najčešće besplatan i već ugrađen u ROM memoriju. Mana mu je, u usporedbi s prevodiocem, izrazita sporost. Tumač je napravljen za programiranje.

Prevodilac je napravljen za izvođenje. Brzina mu je osnovna vrlina. Pouzdanost također. Mane su mu visoka cijena, izdvojenost iz ROM memorije, teško pronalaženje pogrešaka.

	Tumač	Prevodilac
1	NEISKUSAN PROGRAMER	ISKUSAN PROGRAMER
2	NEOBAVEZNA OBRADA	OBAVEZNA (»ŽIVA«) OBRADA
3	ČESTE PROMJENE PROGRAMA	RIJETKE PROMJENE U PROGRAMU
4	KRATKA I RIJETKA OBRADA	DUGA I ČESTA OBRADA
5	POUZDANOST NIJE BITNA	POUZDANOST JE BITNA
6	BRZO PROGRAMIRANJE	POLAKO PROGRAMIRANJE
7	NISKA CIJENA	VISOKA CIJENA

U ovoj tabeli naveli smo osnovne karakteristike koje utječu na odluku o korištenju tumača ili prevodioca.

1. Tumač prevodi svaku liniju na koju naiđe i na taj način možemo jednostavno pratiti izvođenje programa i pronaći pogrešku. Nakon ispravljanja pogreške možemo ponovno izvesti program. Ovakav način ispravljanja pogrešaka pristupačan je početnicima i neiskusnim programerima. Iskusni programeri ili profesionalci navikli su na rad s programskim prevodiocima pa im on ne stvara posebne teškoće.
2. Pojam neobavezne obrade odnosi se na obradu čije rezultate ne očekuje neki vanjski korisnik. Obavezna je obrada koju izvršavamo za nekog korisnika u određenom vremenu i s očekivanim rezultatima obrade. Ovakva vrsta obrade moguća je uistinu jedino pomoću prevodioca.
3. Česte promjene daleko je lakše provesti uz pomoć tumača. Promjena se unese i program se može odmah izvoditi. Ako upotrebljavamo programski prevodilac svaku promjenu treba unijeti pomoću procesora za upisivanje teksta, izvršiti prevođenje i na kraju povezivanje da bismo dobili program koji se može izvoditi. Ovaj cjelokupni postupak može potrajati dosta dugo. Uz upotrebu diskova potrebno je desetak i više minuta.
4. Upotreba programskog tumača je idealna ukoliko obrada traje kratko i ne ponavlja se često. Ako je obrada duža i često se upotrebljava, treba razmotriti mogućnost upotrebe prevodioca. Brzina se u prosjeku povećava 20, a ponekad i više od 100 puta.

5. Ako je pouzdanost bitna, prevodilac nas rješava briga oko neugodnih iznenađenja. Dok tumač može provjeriti tek liniju koja se izvodi, za prevedeni program smo sigurni da nema sintaksičnih pogrešaka.
6. Programiranje pomoću tumača je daleko brže nego pomoću prevodioca jer je cjelokupna logika upravo tako i izvedena.
7. Dok je tumač najčešće ugrađen u opremu, prevodioci stoje u prosjeku od 200 do 300 funti.

Najzgodnija je kombinacija tumaća i prevodioca (moguće za programske proizvode firme Microsoft) koji su međusobno prilagodljivi. Za izradu, razvoj i testiranje programa možemo upotrebljavati tumač, a za kasnije stalno upotrebljavanje programa njegovu prevedenu verziju.

No, bez obzira na međusobnu usklađenost postoje neke razlike u upotrebi koje se odnose na rad s tabelama:

DIM TABELA$(50)

se može upotrebljavati za tumač i prevodilac, dok prevodilac ne razumije:

DIM TABELA$(J%)

Treba paziti da se pri upotrebi prevodioca ne koriste složene sistemske DEF naredbe, kao niti ostali složeni izrazi, jer postoji mogućnost da ih prevodilac odbaci. Treba pripaziti i da se FOR-NEXT i WHILE-WEND petlje nalaze u fizičkom slijedu.

Osim za zabavu BASIC može služiti i za izradu ozbiljnih aplikacija. U tom slučaju preporuća se rad s prevodiocem. Vjerojatno će se uloženi novac (ako je uopće uložen) ubrzo vratiti i isplatiti.

Željko Pažur

NAJZNAČAJNIJI PROIZVOĐAČI BASIC prevodioca

Ovdje su navedeni samo neki (najpoznatiji) proizvođači BASIC prevodioca. Cijene su približne i mogu se razlikovati od ovdje navedenih.

MICROSOFT BASIC COMPILER
Vrste: BASIC-80, BASIC-86
Prilagodljivi: MBasic i ostali Microsoft programski tumači
Upravljački sistem: CP/M-80, MS-DOS, PC-DOS, CP/M-86.
Cijena: 275 funti

DIGITAL RESEARCH BASIC-2
Vrsta: CBasic-2
Prilagodljiv: DIGITAL RESEARCH BASIC COMPILER
Upravljački sistem: CP/M
Cijena:8-bitna verzija – 100 funti
16-bitna verzija – 200 funti

DIGITAL RESEARCH BASIC COMPILER
Vrsta: DIGITAL RESEARCH BASIC COMPILER CB-80, CB-86
Prilagodljiv: CBasic-2
Upravljački sistem: Sve verzije CP/M. Podržava istovremeni rad s više korisnika (multi-user support)
Cijena: 320 funti

SUPERSOFT BASIC COMPILER
Prilagodljiv: Verzije Microsoft Basica
Upravljački sistem: MS-DOS, PC-DOS, CP/M-86 ali ne i 8-bitni upravljački sistemi.

Najjača osobina raznih BASIC-a tvrtke Microsoft je njihova međusobna prilagodljivost, što omogućuje pisanje i testiranje programa tumačem, i nakon prevođenja izvođenje apsolutnog programa. CBasic-2 se nalazi na tržištu od 1978. Prevodi vrlo brzo, ali proizvodi međuprogram koji tumać mora najprije objasniti, pa je izvođenje sporije od programa prevedenih pravim prevodiocima. Ista tvrtka proizvodi prevodioce CB-80 i CB-86 za 8 i 16-bitne sisteme. Napravljen je i prevodilac Microsoft CBasic koji se zove M2CBasic pod kontrolom upravljačkog sistema CP/M-86.

Tvrtka Supersoft proizvela je 16-bitni prevodilac koji može (kako tvrde proizvođači), prevoditi sve Microsoft-Basic programe. Primjenom binarne aritmetike izbjegavaju se problemi zaokruživanja koji se pojavljuju u aritmetici pomičnog zareza visoke preciznosti.

Željko Pažur





Programska biblioteka za početnike

SPECTRUM

UVOD

Kod prijatelja ste vidjeli mikroračunalo na djelu; oduševile su vas mogućnosti pa ste poželjeli i sami postati vlasnik i korisnik sličnog uređaja. Stručne i pomalo nedokučive manipulacije tastaturom i kazetofonom, koje su također bile dio prikazivanja rada računala, nisu vas obeshrabrile i – da ne duljimo – nabavili ste vlastitu opremu ili ste na najboljem putu da to uradite.

Kao neiskusni korisnik mikroračunala imate na raspolaganju popratnu literaturu (upute za rad) i programsku kazetu za zorno objašnjenje rada na računalu, te nekoliko kupljenih ili posuđenih programskih kazeta s igrama ili drugim programima. Pretpostavljamo da odgovarajući kazetofon već posjedujete (posve zadovoljavajući su standardni modeli tzv. reporterskih kazetofona; radiokazetofon također potpuno zadovoljava i ima određene prednosti o kojima će biti riječi drugom prilikom). TV primjenik je također spreman. Preporučujemo korištenje manjih prenosivih uređaja i to iz posve praktičnog razloga da biste mogli raditi neometano i uz punu koncentraciju.

Pročitali ste upute za rad (ili barem onaj njihov dio o priključivanju i međusobnom povezivanju računala i odgovarajućih uređaja) te upute za korištenje »demo« programske kazete. Razumljivo je da jezična zapreka s jedne i brojnost stručnih izraza s druge strane predstavljaju priličnu poteškoću za potpuno razumijevanje svih detalja iz literature, pretpostavljamo da ste ipak (makar smo na osnovu skice spajanja pojedinih uređaja, koja je obavezno navedena), izvršili sva potrebna priključivanja i uspjeli ostvariti punjenje (LOAD) prvog programa sa »demo« programske kazete.

Ukoliko ste ipak prilikom međusobnog povezivanja pojedinih dijelova opreme naišli na nepremostive poteškoće, kao npr. nepodudarnost tipa priključaka za kazetofon ili TV prijemnik, odgovarajuća rješenja navesti ćemo u posebnom prilogu.

Rad s »demo« programskom kazetom je prilično jednostavan: nakon uspješnog aktiviranja prvog programa potrebno je samo slijediti upute s ekrana. Jezična i terminološka barijera su i nadalje glavne poteškoće za potpuno razumijevanje izlagane materije, no uvelike su nadvladane naglašenom vizuelnom komponentom (razumljivom slikom). Slična situacija je i u pogledu korištenja drugih programskih kazeta, s komercijalnim igrama ili drugim programima. Nakon uspješnog punjenja početka (prvog dijela) programa potrebno je samo slijediti upute s ekrana. Međutim, ovaj najjednostavniji nivo korištenja mikroračunala ne može niti izdaleka zadovoljiti zainteresiranijeg korisnika; s druge strane korištenje komercijalnih programa bez mogućnosti stvaranja vlastitih kopija, odnosno biblioteke programa, predstavlja značajni financijski izdatak.

U daljim poglavljima bit će navedeni osnovni principi, postupci i programska rješenja koji korisniku mikroračunala omogućuju stvaranje vlastite programske biblioteke, prijepis komercijalnih i drugih programa s posuđenih na vlastite programske kazete, te vođenje osnovne programske dokumentacije.

Obzirom da su pojedina programska rješenja specifična samo za određeni tip mikroračunala, opredijelili smo se za nesumljivo najpopularniji i najzastupljeniji model mikroračunala koji se koristi u našoj zemlji – Sinclair ZX Spectrum. Ovo nipošto ne znači da korisnici mikroračunala drugih proizvođača neće u daljim izlaganjima naći korisne i zanimljive pojedinosti.

OSNOVNI POJMOVI

Programska biblioteka je uređena skupina programa spremnih za korištenje. Nije potrebno posebno naglašavati da je raspolaganje obimnom i dobro uređenom programskom bibliotekom osnova za uspješan razvoj vlastitih rješenja, odnosno za svakodnevno korištenje mikroračunala na način koji nam najviše odgovara. U informatici općenito, dakle u svijetu »velikih« informacijskih sistema, programske biblioteke se još nazivaju i programska podrška, a snaga i vrijednost pojedinog sistema redovito se procijenjuju na osnovu obima, mogućnosti i karakteristika postojećih i dostupnih programskih proizvoda.

Programsku biblioteku za mikroračunalo sačinjavaju programske kazete sa snimljenim programima, liste sadržaja pojedinih kazeta, kao i abecedni popis programa, te upute za korištenje svakog programa te eventualno liste izvornih naredbi pojedinih programa (source listing).

Program je uređeni niz naredbi razumljivih računalu koje se nalaze na mediju pristupačnom računalu (npr. magnetska vrpca) i čijim se izvršavanjem na računalu ostvaruje željeni cilj. Pisanje novog programa je stvaralački čin pri čemu autor ispisuje odgovarajuće programske naredbe, odnosno unosi ih u računalo putem tastature. Dakle, program može postojati i u obliku liste izvornih naredbi. Ovakva izvedba programa je nepogodna za neposredno korištenje (potrebno je utipkavanje) i upotrebljava se isključivo kod razvijanja novih rješenja. Uz svaki program postoje odgovarajuće upute za korištenje koje mogu biti navedene u samom programu ili postojati kao samostalni dio programske dokumentacije.

Programi se mogu podijeliti na razne načine, ovisno o promatranim svojstvima. U našem slučaju značajno je poznavanje fizičkog ustrojstva programa, kako bismo ga mogli uspješno spremiti ili prepisati.

Kod mikroračunala ZX Spectrum osnovna programska struktura na programskoj kazeti redovno se sastoji iz dva dijela koje ćemo nazvati blok zaglavlja i blok sadržaja. Punjenje programa sa programske kazete u memoriju računala najjednostavnije se ostvaruje naredbom LOAD "" (što doslovno znači: »napuni slijedeći program na koji naiđeš pri čitanju programske kazete u memoriju«), a aktiviranje napunjenog programa najčešće se ostvaruje utipkavanjem naredbe RUN.

Postoje naravno i drugi načini punjenja programa s programske kazete u memoriju, kao i drugi načini aktiviranja napunjenog programa, uključujući i automatsko aktiviranje, no o tim specifičnim slučajevima bit će riječi drugom prilikom. Prilikom punjenja, blok zaglavlja nalazi se ispred bloka sadržaja, relativno je kratak i nakon njegovog punjenja na ekranu se pojavljuje naziv programa. Nakon bloka zaglavlja blok sadržaja puni se automatski i nije potrebno ponovo utipkavati naredbu LOAD. Sve ove detalje mogli smo uočiti u radu s »demo« programskom kazetom, a isto vrijedi i za svaki BASIC program koji sami utipkamo u računalo. Nakon spremanja na programsku kazetu naredbom SAVE "ime-prog" i premotavanja kazete na početak, možemo nakon ponovnog punjenja naredbom LOAD "" uočiti da program ima navedenu strukturu.

Određeno proširenje pravila o ustrojstvu programa predstavljaju složeni programi. Takvi su redovito svi komercijalni programi (npr. igre) i njihovo je svojstvo da se sastoje iz nekoliko povezanih dijelova – osnovnih programa. Povezanost osnovnih programa unutar složenog programa očituje se u tome da se nakon naredbe LOAD "" svi dijelovi programa pune automatski. Pri tom je važno napomenuti da, ovisno o funkcijama pojedinih osnovnih programa, uobičajeno prikazivanje naziva pojedinih dijelova na ekranu može izostati.

U svakodnevnom radu pojam programa se odnosi podjednako na složene i na osnovne programe, o čemu u svakom slučaju treba voditi računa, posebno prilikom formiranja popisa programa programske biblioteke.

Navedena pravila o ustrojstvu programa imaju i izuzetke. Radi se o tzv. programima bez zaglavlja (headerless). Pri tom u sklopu nekog složenog programa mogu biti osnovni programi sa i bez zaglavlja, o čemu treba voditi računa – posebno prilikom kopiranja. Obzirom na relativno malu učestalost programa bez zaglavlja, o njihovom ustrojstvu i osobinama pisat ćemo drugom prilikom.

PRAKTIČNI PRIMJERI

U ovom dijelu objasniti ćemo detaljno tri jednostavna programa s preciznim uputama za arhiviranje i korištenje. Radi se o programu za testiranje ispravnosti memorije ZX Spectruma, programu za kopiranje programa, te o programu za ispisivanje sadržaja programske kazete na ekran. Svaki program je predstavljen u obliku liste izvornih naredbi koje se mogu direktno utipkati u računalo. Nakon toga preostaje samo aktiviranje programa naredbom RUN, odnosno spremanje (arhiviranje) na programsku kazetu odgovarajućom naredbom SAVE. Pri korištenju ovih programa koristi se minimalno rad u strojnom kodu (preko DATA, POKE, RANDOMIZE USR i PEEK BASIC naredbi). Svaki program moguće je prekidati u izvođenju standardnom naredbom BREAK putem tastature i ponovno aktivirati novom naredbom RUN. Prema ustrojstvu svaki program predstavlja standardni osnovni program sa zaglavljem i može ga se jednostavno prepisati na programske kazete. Svaki je program pisan u sažetom obliku, bez REM linija i s po nekoliko naredbi u pojedinoj programskoj liniji. To je uči-

njeno radi što bržeg i jednostavnijeg unošenja programa u računalo i što manjeg zauzeća memorije (kod programa za kopiranje ovaj je zahtjev posebno važan). Nije međutim isključena mogućnost naknadnog proširenja svakog od programa dodatnim komentarima, odnosno literalima, što prepuštamo ambicioznijim korisnicima.

PROGRAM MR-TEST

Program služi za testiranje slobodnog dijela korisničke memorije mikroračunala ZX Spectrum (oba modela: 16K i 48K). U jednoj jedinoj programskoj liniji nalazi se naredba DATA sa 30 podataka (dekadski iskazan strojni kôd procesora Z80), zatim FOR-NEXT grupa s naredbama READ i POKE, kojima se strojni kôd unosi u memoriju računala od apsolutne adrese 23296 do apsolutne adrese 23325, te naredba PRINT USR 23296 kojom se spomenuti strojni program aktivira.

Koristeći se standardnim postupcima za unos i ispravljanje podataka pažljivo utipkamo program sa liste i nakon toga utipkamo naredbu RUN. Za koji trenutak ekran će se isprazniti, da bi se nakon toga u gornjem desnom uglu pojavio peteroznamenkasti broj. Za model 16K broj treba imati iznos 32575, a za model 48K iznos 65343. To znači da su sve slobodne memorijske lokacije ispitane na način da se u sve bitove svakog bajta najprije upiše, pa pročita binarna nula, a nakon toga se isti postupak ponovi s binarnim jedinicama. U slučaju dobivanja nekog drugog broja ili totalnog zastoja računala, moguča su dva uzroka: greška kod utipkavanja programa ili greška memorije računala.

Ovaj program je izuzetno koristan kod testiranja ili ispravljanja kompleksnih programa (služi za otklanjanje sumnje u ispravnost računala u delikatnim situacijama), te kod kupovine novog ili rabljenog računala za kontrolu ispravnosti uređaja.

ARHIVIRANJE PROGRAMA

Kada se dobivanjem ispravnih rezultata osvjedočimo da je utipkani program ispravan, odnosno da naše računalo radi besprijekorno, možemo program pohraniti na programsku kazetu. Utipkamo naredbu SAVE "MR-TEST" i nakon ENTER i poruke "Start tape, then press any key." iskopčamo EAR utikač na priključenom kazetofonu, uključimo RECORD-PLAY i utipkamo bilo koji znak. Kretanje vodoravnih linija po okviru slike na ekranu pokazuje da je spremanje na programsku kazetu u toku. Pojava poruke "0 OK, 0:1" znači da je snimanje na programsku kazetu gotovo, pa nakon cca jedne sekunde (radi dobivanja razmaka između pojedinih zapisa na kazeti) isključimo (STOP) kazetofon.

Provjeru ispravnosti spremljenog programa možemo izvršiti postupkom verifikacije (prema uputama za rad dobivenim uz računalo), ili jednostavno punjenjem upravo spremljenog programa u računalo. U tom slučaju očistimo memoriju računala (iskopčavanjem i ponovnim ukopčavanjem utikača napajanja), premotamo kazetu na početak i utipkamo naredbu LOAD "", nakon



čega ukopčamo EAR utikač u priključeni kazetofon i uključimo PLAY. Kretanje vodoravnih linija po okviru slike na ekranu ponovo pokazuje da je punjenje u toku. Nakon punjenja zaglavlja, na ekranu se pojavljuje poruka "Program: MR-TEST", da bi se na kraju punjenja pojavila uobičajena poruka "0 OK, 0:1". Nakon toga preostaje da utipkamo naredbu RUN i utvrdimo da je program ispravno izvršen, što je ujedno i dokaz da je zapis na programskoj kazeti ispravan.

Ovakav način provjere ispravnosti arhiviranog programa redovno se koristi prilikom testiranja kopija programa, kad je standardni postupak verifikacije nemoguće provesti (radi automatskog startanja, originala u strojnom kodu i sl.).

Drugi način spremanja programa na programsku kazetu ostvaruje automatsko aktiviranje programa nakon svakog naknadnog punjenja s kazete u računalo. Razlika je jedino u naredbi SAVE koja u ovom slučaju ima slijedeći oblik SAVE "MR-TEST" LINE 1. To znači da se prilikom svakog punjenja, nakon završetka punjenja program počinje automatski izvršavati naredbama navedenim u programskoj liniji broj 1 (dakle od početka). Postupak provjere ispravnosti spremljenog programa izvršavamo na isti način kao i u prijašnjem slučaju, uz napomenu da nakon naredbe LOAD "" izostaje naredba RUN, jer program započinje automatski. Ovakav način spremanja programa karakterističan je za veliku većinu komercijalnih programa, pa ga preporučamo za spremanje vlastitih programa, čime nesumnjivo povećavamo efikasnost rada računala.

Postupak spremanja novog programa na programsku kazetu, kao i postupak punjenja programa s programske kazete u računalo pregledno i sažeto je prikazan tabelom. Preciznim izvođenjem svakog koraka postupka osiguravamo u najvećoj mjeri efikasnost rada računala, kao i sigurnost podataka kojima rukujemo (programi).

PROGRAMSKA BIBLIOTEKA

Prije nastavka prikaza opisom slijedećeg programa posvetit ćemo nekoliko napomena programskim kazetama. U opisu spremanja programa MR-TEST na programsku kazetu, pretpostavili smo da se radi o novoj, dakle praznoj, kazeti, na početak koje smo snimili program, odnosno s početka koje smo – nakon premotavanja – punili spremljeni program u računalo. Spremanjem programa na tu kazetu stvorena je osnova buduće biblioteke programa, spomenuti program je tek prvi od budućeg mnoštva i trebalo bi ga na neki način unijeti u popis programa.

Popis programske biblioteke se u najjednostavnijem slučaju sastoji iz dvije vrste popisa: jedno su liste sadržaja svake programske kazete, a drugo je cjelovit abecedni popis svih raspoloživih programa. U našem slučaju listu sadržaja programske kazete možemo upisivati na za to predviđeni dio omota kazete s ucrtanim rubrikama ili na odgovarajući list papira koji uložimo u kutiju kazete. Obzirom na mogućnost snimanja s obje strane kazete svaki ovakav popis sastojat će se iz dva dijela A i B, dok svaku pojedinu kazetu koju koristimo možemo označiti prikladnim nazivom ili brojem. Podaci koje unosimo za pojedini program su, pored naziva, redni broj i broj brojača vrpce (Tape Counter) s kazetofona na kojem počinje program. Osim ovih rubrika u popis je korisno unijeti i podatke o veličini programa u bajtovima, tip programa (BASIC, strojni kôd), način spremanja na programsku kazetu (SAVE, kopiranje), apsolutnu adresu početka (za programe u strojnom kodu), datum spremanja programa (i eventualno porijeklo originala), te, konačno, status programa (izvodljiv ili ne). Za sveobuhvatni abecedni popis svih raspoloživih programa najjednostavnije je upotrijebiti neki neiskorišteni adresar. Pored naziva programa, u ovaj popis ulazi još samo oznaka programske kazete na kojoj se program nalazi. Važno je još napomenuti da se u liste sadržaja programskih kazeta složeni programi upisuju dio po dio prema stvarnim nazivima iz pojedinih zaglavlja, dok se u abecednom popisu programi evidentiraju pod komercijalnim imenima.

Radi jednostavnijeg arhiviranja pojedinih programa na posve određene programske kazete (spremanje programa prema tipu: npr. posebno akcione igre, posebno avanture itd.) u praksi se osim arhivskih kazeta (koje su redovito većeg kapaciteta, npr. C-60 ili C-90), koriste i tzv. radne kazete koje su manjeg kapaciteta i obično sadrže nekoliko najvažnijih uslužnih programa (npr. program za testiranje memorije, program za kopiranje itd.), odnosno radne verzije programa koje upravo kreiramo. Na ovaj način omogućen je brži i efikasniji rad računala, a programska biblioteka je sređenija.

Pretpostavimo da je program MR-TEST koji smo upravo spremili snimljen na radnu kazetu. Navodimo postupak za njegovo kopiranje na arhivsku kazetu.

PROGRAM MR-COPY

Program služi za kopiranje osnovnih programa sa zaglavljima s originalne programske kazete na neku drugu programsku kazetu. To znači da se složeni programi (a to je redovit slučaj kod npr. komercijalnih igara) kopiraju dio po dio.

Linije 1 i 2 sadrže dekadski interpretiran strojni kôd, te FOR-NEXT grupu za unos strojnog koda u memoriju. Linije 3, 4 i 5 sadrže naredbe za punjenje originala u memoriju, te ispisivanje naziva i vrste programa na ekranu, dok se naredbama linije 6 ispisuje veličina originalnog programa. Konačno linija 7 sadrži naredbe vezane uz spremanje kopije na drugu programsku kazetu.

Nakon pažljivog utipkavanja naredbi s liste program je spreman za upotrebu. Obzirom da prilikom izvođenja program koristi dvije programske kazete (original i kopija) moramo biti izuzetno pažljivi u radu, kako nepažnjom ne bi došlo do brisanja originalnih podataka (programa). Da posve izbjegnemo mogućnost ovakove pogreške, korisno je na svim kazetama prije upotrebe odstraniti zaštitne pločice na stražnjem dijelu kazete, čime se onemogućuje snimanje – dakle i brisanje – zapisa s kazete. Naknadno lijepljenje komadića samoljepljive vrpce preko spomenutog dijela kazete privremeno će omogućiti snimanje na kazetu kada to bude potrebno. Ovakva praksa zaštite podataka na trakama uobičajena je kod velikih računala i nema razloga da je ne primijenjujemo i na malim računalima.

Postupak testiranja programa MR-COPY donekle se razlikuje od testiranja prijašnjeg programa. Obzirom da testiranje kopije pretpostavlja brisanje samog programa za kopiranje iz memorije,

najprije ćemo izvršiti spremanje utipkanog programa na radnu programsku kazetu naredbom SAVE "MR-COPY" LINE 1, provođenjem postupaka navedenih u poglavlju o spremanju programa. Tek tada pristupamo testiranju programa za kopiranje.

KOPIRANJE PROGRAMA

Pretpostavimo da želimo programe MR-TEST i MR-COPY, koje smo nakon utipkavanja naredbama SAVE spremili na radnu programsku kazetu iskopirati s te kazete na neku drugu, arhivsku kazetu.

Jedna od mogućnosti kopiranja, koju samo uzgred spominjemo, je klasično »audio« presnimavanje, za koje nam treba drugi kazetofon i odgovarajući provodnik (»kabel«) za povezivanje te sreća da se svaki bit originala pravilno preslika na kopiju. Naime, utjecaj smetnji (šuma) kod ovakvog presnimavanja je vrlo izražen, pa je čest slučaj da kopije budu nečitljive za računalo. Prilikom ovakvog kopiranja izostaju sve važne informacije o presnimljenom materijalu (naziv programa, veličina itd.), što prilično otežava svrstavanje u programsku biblioteku.

Druga mogućnost kopiranja programa je punjenje originala u računalo na standardni način naredbom LOAD, te nakon toga spremanje na kopiju odgovarajućom naredbom SAVE. Prvi nedostatak ove metode je što programe koji se automatski aktiviraju nakon punjenja moramo prisilno prekidati naredbom BREAK (što može imati neželjene posljedice), dok je drugi nedostatak što veliku većinu komercijalnih programa uopće ne možemo prekinuti naredbom BREAK.

Standardni način presnimavanja programa je tzv. programsko presnimavanje. Najprije u računalo napunimo posebni program za presnimavanje, pod čijom kontrolom se nakon toga vrši punjenje originalnog programa s jedne programske kazete i njegovo spremanje kao kopije na drugu programsku kazetu. Program za presnimavanje može osim izvođenja kopiranja ispisati podatke o nazivu, vrsti i veličini kopiranog programa.

Program MR-COPY predstavlja jedan takav posebni program. Nakon spremanja na kazetu, koje smo u međuvremenu obavili, on je i nadalje u memoriji računala i spreman je za testiranje. Najprije premotamo kazetu (na kojoj se nalaze programi MR-TEST i MR-COPY na početak (odnosno na početak programa MR-TEST), te ujedno pripremimo još jednu praznu kazetu na koju ćemo izvršiti kopiranje. Prvu kazetu ćemo nazivati original, a drugu kopija. Radi veće sigurnosti potrebno je kazeti original odstraniti pločice na stražnjoj strani, čime je zaštićujemo od brisanja. Isto tako trebamo omogućiti snimanje na kazetu kopija lijepljenjem komadića samoljepive vrpce na mjesta s kojih su spomenute pločice eventualno prije odstranjene. Kazetofon je priključen na uobičajeni način (EAR i MIC su uključeni), postavimo kazetu original i utipkamo naredbu RUN. Ako je program MR-COPY korektno utipkan u računalo, samo slijedimo upute koje se nižu jedna iza druge. Ukoliko se pojave odstupanja, treba program prekontrolirati i ponoviti postupak nakon otklanjanja otkrivenih grešaka.

Naredba RUN izazvat će ispis poruke UCITAVANJE ORIGINALA uz kratki podsjetnik o predstojećim aktivnostima. Uključimo PLAY i kretanje vodoravnih linija po okviru slike na ekranu pokazat će da je punjenje originala u toku. Nakon punjenja bloka zaglavlja i bloka sadržaja originalnog programa, na ekranu se pojavljuje oznaka vrste napunjenih podataka (program ili bytes ukoliko se radi o strojnom kodu), naziv, te veličina izražena u bajtovima. Poruka UPISIVANJE KOPIJE uz kratki podsjetnik o slijedećim aktivnostima označuje da je sve spremno za spremanje kopije. Zaustavimo kazetofon i isključimo EAR utikač. Podatke o programu upišemo u odgovarajuće evidencije programske biblioteke. Izvadimo kazetu original i postavimo kazetu na koju želimo kopirati. Nakon toga utipkamo ENTER i nakon poruke "Start tape, then press any key". uključimo RECORD-PLAY na kazetofonu. Dalji postupak je sličan standardnom postupku pri spremanju novih programa, jedina razlika je da se nakon završetka spremanja, program MR-



COPY ponovo automatski počinje izvoditi od početka, čime se čitav postupak nastavlja učitavanjem novog originala. Izvršimo i ovo kopiranje na već opisani način i nakon ponovnog aktiviranja programa prekinemo dalje izvršavanje utipkavanjem naredbe BREAK. Sada nakon brisanja memorije računala možemo provjeriti ispravnost upravo načinjenih kopija programa MR-TEST i MR-COPY, tako da ih punimo u računalo i zatim kontroliramo njihovo izvršavanje.

Postupak kopiranja programa primjenom programa MR-COPY sažeto je prikazan tabelom. Pažljivo izvršavanje svakog koraka postupka kopiranja, ma kako na prvi pogled složeno djelovalo, vrlo brzo postaje posve rutinsko. Eventualno uključivanje drugog kazetofona još više pojednostavljuje i ubrzava postupak. Na ovaj način programe koje ste do jučer kupovali ili posuđivali, sada možete razmjenjivati, pri čem se obim vaše programske biblioteke neprestano povećava. Tokom kopiranja posuđenih programa na vlastite kazete, prije ili kasnije, naići ćete na programsku kazetu s nepotpuno naznačenim sadržajem. Analiza takve kazete s koje želite presnimiti najatraktivnije programe često može oduzeti dosta vremena, no i tome se može doskočiti. Program koji navodimo jedno je od mogućih rješenja ovog problema.

```
MR-TEST

   1 DATA 42,101,92,68,77,62,0,1
19,126,254,0,192,62,255,119,126,
254,255,192,167,35,237,114,200,1
67,237,122,3,24,231: FOR n=23296
 TO 23325: READ m: POKE n,m: NEX
T n: PRINT USR 23296

lista programa MR-TEST
```

Postupak spremanja programa iz memorije računala na programsku kazetu i punjenja programa s kazete u računalo

1. računalo je uključeno, TV prijemnik je uključen
   priključak MIC uključen u kazetofon
   priključak EAR isključen iz kazetofona
   programska kazeta spremna za snimanje
2. utipkamo novi program s liste u računalo
3. utipkamo SAVE "ime-prog" ili SAVE "ime-prog" LINE n utipkamo ENTER
4. poruka Start tape, then press any key.
5. uključimo RECORD-PLAY na kazetofonu utipkamo bilo koji znak
6. poruka 0 OK, 01
   isključimo (STOP) kazetofon nakon cca 1 sek.
7. premotamo kazetu na početak
   uključimo EAR priključak u kazetofonu
   utipkamo LOAD "" (ili LOAD "" CODE za poseban slučaj)
   uključimo PLAY na kazetofonu
8. poruka 0 OK, 01 ili automatski početak programa
   isključimo (STOP) kazetofon
   utipkamo RUN (ako se program nije automatski započeo izvršavati)

Postupak kopiranja programa s kazete original na kazetu kopija izvođenjem uslužnog programa MR-COPY

1. računalo je uključeno, TV prijemnik je uključen
   priključak MIC uključen u kazetofon
   priključak EAR uključen u kazetofon
   programska kazeta s programom MR-COPY spremna za izvođenje
2. utipkamo LOAD "" i ENTER
   uključimo PLAY na kazetofonu
3. poruka Program: MR-COPY (nestaje nakon punjenja)
   poruka UCITAVANJE ORIGINALA
   kazeta = original, uključeno EAR, PLAY
   isključimo (STOP) kazetofon
   postavimo kazetu original spremnu za izvođenje
   uključimo PLAY na kazetofonu
4. poruka Program: ime-prog nnn
   UPISIVANJE KOPIJE
   kazeta = kopija, isključeno EAR, ENTER
   isključimo (STOP) kazetofon
   isključimo EAR priključak iz kazetofona
   postavimo kazetu kopija spremnu za snimanje
   utipkamo ENTER
5. poruka Start tape, then press any key.
6. uključimo RECORD-PLAY na kazetofonu
   utipkamo bilo koji znak
7. poruka UCITAVANJE ORIGINALA
   kazeta = original, uključeno EAR, PLAY
   isključimo (STOP) kazetofon
   uključimo EAR priključak u kazetofon
   postavimo kazetu original spremnu za izvođenje
   uključimo PLAY na kazetofonu
   nastavimo postupak prema točki 4.

```
MR-COPY

   1 DATA 221,33,0,91,33,168,97,
205,112,9,201,221,33,0,91,17,17,
0,62,0,55,205,2,8,221,33,0,91,22
1,86,12,221,94,11,221,33,168,97,
62,255,55,205,2,8,201
   2 FOR n=23315 TO 23359: READ
a: POKE n,a: NEXT n: CLEAR 24999
   3 PRINT AT 1,5;"UCITAVANJE OR
IGINALA": PRINT AT 3,1;"kazeta=o
riginal, uklj.EAR, PLAY": RANDOM
IZE USR 23326: IF PEEK 23296=0 T
HEN PRINT AT 9,1;"Program:": GO
TO 5
   4 PRINT AT 9,3;"Bytes:"
   5 FOR n=23297 TO 23306: IF (P
EEK n>31 AND PEEK n<165) THEN PR
INT AT 9,(n-23287);CHR$ PEEK n;:
 NEXT n
   6 PRINT AT 9,25;PEEK 23307+25
6*PEEK 23308: POKE 23629,255
   7 PRINT AT 17,5;" UPISIVANJE
KOPIJE":PRINT AT 19,0;"kazeta=
kopija, isklj.EAR, ENTER": PAUSE
 0: CLS : RANDOMIZE USR 23315: G
O TO 3

lista programa MR-COPY
```

```
MR-TTOC

  10 CLS : PRINT AT 7,0;"LISTA S
ADRZAJA PROGRAMSKE KAZETE": PRIN
T AT 20,2;"LOAD TAPE, ENTER, PLA
Y...": PAUSE 0: CLS : RESTORE
  15 FOR n=23296 TO 23308: READ
m: POKE n,m: NEXT n: DATA 175,55
,221,33,104,91,17,17,0,205,86,5,
201
  20 LET i=1: PRINT ;"BR.  N A Z
 I V    VRSTA  DULJINA"
  25 RANDOMIZE USR 23296: IF PEE
K 23400>3 THEN GO TO 25
  30 LET j=i+1
  35 IF j<22 THEN GO TO 45
  40 j=j-20: GO TO 35
  45 PRINT AT j,0;"
                     ": PRINT AT j
,0;i
  50 FOR n=23401 TO 23410: IF (P
EEK n>31 AND PEEK n<165) THEN PR
INT AT j,(n-23396);CHR$ PEEK n;:
 NEXT n
  55 IF PEEK 23400=0 THEN PRINT
AT j,17;"Program": GO TO 65
  60 PRINT AT j,17;"Bytes *"
  65 PRINT AT j,26;PEEK 23411+25
6*PEEK 23412: POKE 23400,255: PO
KE 23692,255: LET i=i+1: GO TO 2
5

lista programa MR-TTOC
```

PROGRAM MR-TTOC

Program služi za ispisivanje sadržaja programske kazete na ekranu. Pri tom se svaki osnovni program na kazeti tretira kao posebna cjelina, pa se na listi složeni programi pojavljuju u sastavnim dijelovima (što olakšava kopiranje).

Linije 10 i 15 sadrže početne poruke programa, kao i dekadski iskazan strojni kôd, te FOR-NEXT grupu naredbi za unos strojnog koda u memoriju računala. Linija 20 ispisuje zaglavlje liste, dok glavni programski ciklus započinje linijom 25. Rutine za ispisivanje naziva, vrste i veličine programa slične su odgovarajućim rutinama programa MR-COPY. Važno je napomenuti da se nakon ispunjavanja ekrana podacima (nakon 20 učitanih osnovnih programa) ne ostvaruje pomak liste na gore (scroll) već se slijedeća linija upisuje na mjesto prve itd.

Nakon pažljivog utipkavanja naredbi s liste, program je spreman za korištenje. Dovoljno je samo postavit programsku kazetu u priključeni kazetofon i započeti program utipkavanjem naredbe RUN, te uključivanjem PLAY na kazetofonu.

Detalji spremanja programa na programsku kazetu i ponovno punjenje u računalo nimalo se ne razlikuju od do sada navedenih postupaka, pa ih nećemo posebno navoditi.

Damir Fio



Editor za definiranje funkcijskih tipki – BBC

Mikroračunalo BBC dobro je poznato svima koji prate domaće i strane časopise iz svijeta mikroračunala, te stoga nije potrebno posebno isticati sve njegove kvalitete i karakteristike. Jedna od njegovih kvaliteta, mogućnost definiranja funkcijskih tipki, korisniku može znatno olakšati rad s ovim računalom. Namjera ovog članka je da pomogne korisnicima da što bolje iskoriste funkcijske tipke.

FUNKCIJSKE TIPKE

Funkcijske tipke lako su uočljive na tastaturi jer se bojom razlikuju od ostalih tipki. Označene su slovima f0 do f9. Njihova upotreba mogla bi se ukratko opisati na slijedeći način: ukoliko se nekoj od funkcijskih tipki pridruži jedna ili više naredbi, tada se

pritiskom na tu tipku definirana sekvenca može pozvati u bilo kojem trenutku, kad računalo očekuje neki unos s tastature.
Prvo, što se nameće kao praktična definicija za funkcijske tipke su jednostavne naredbe (npr. RUN, LIST, OLD, AUTO, RENUMBER i sl.). Treba, međutim, naglasiti da se svakoj funkcijskoj tipki može pridružiti i čitav niz naredbi, što predstavlja njihovu najznačajniju karakteristiku i otvara široki spektar mogućnosti njihovog korištenja.

EDITOR I NJEGOVE FUNKCIJE

Editor tj. program za definiranje funkcijskih tipki, čiji ispis slijedi u nastavku, ima nekoliko funkcija, koje korisniku omogućuju da na jednostavan način definira tri skupa (tj. 3*9) različitih funkcijskih tipki. Pored toga, omogućuje trajno spremanje definicija i njihovo učitavanje kod ponovnog uključivanja računala, ispisivanje definicija na ekran, te grafički prikaz naziva funkcijskih tipki. Nazive funkcijskih tipki korisnik sam određuje, a služe kao podsjetnik na funkciju koju obavlja pojedina tipka.
Kad se program jednom pokrene, pritiskom na tipku TAB može se u svakom trenutku pozvati Editor, što se manifestira postavljanjem u grafički mod rada 0, ispisivanjem naziva skupa funkcijskih tipki koji je u tom trenutku aktivan, te ispisivanjem funkcija Editora. Izvođenje bilo koje od funkcija Editora postiže se upisom broja koji je na popisu funkcija pridružen željenoj funkciji (brojevi od 1 do 4). Treba napomenuti da se kod izbora svake slijedeće funkcije Editora mora (pomoću tipke TAB) najprije pozvati popis funkcija.
Prva funkcija omogućuje ispisivanje definicija trenutno aktivnog skupa funkcijskih tipki.
Druga funkcija omogućuje redefiniranje trenutno aktivnog skupa funkcijskih tipki. Nakon što se izabere ova funkcija, potrebno je upisati broj funkcijske tipke koja se želi redefinirati (odnosno K za kraj redefiniranja) i nakon toga njenu definiciju. Ovdje je važno napomenuti da je funkcijska tipka f0 rezervirana za proceduru koja vrši cikličku izmjenu skupova definicija funkcijskih tipki (procedura SWAP) i Editor ne dopušta njeno redefiniranje. Ukoliko je aktivan grafički mod 0, pritiskom na funkcijsku tipku f0 se ujedno grafički prikazuju i nazivi definicija funkcijskih tipki u gornjem dijelu ekrana.
Treća i četvrta funkcija omogućuju spremanje odnosno učitavanje definicija. Kod ovih funkcija potrebno je upisati ime pod kojim će se definicije spremati odnosno prema kojem će se učitavati.

OPIS PROGRAMA PO LINIJAMA I PROCEDURAMA

10	Omogućava definiranje tipke TAB kao funkcijske tipke 10
20	Inicijalizacija pomoćnih spremnika za definiranje funkcijskih tipki (Napomena: memorijsko područje od &900 do &AFF ovdje se koristi za pomoćne spremnike jer ga većina programa ne koristi)
30-70	Strojni program koji ciklički zamjenjuje sadržaje pomoćnih spremnika sa sadržajem spremnika za funkcijske tipke (&B00 do &BFF) – lociran je od &CCO (Napomena: memorijsko područje od &COO do &CFF koje je namijenjeno za definiranje korisničkih simbola koristi se u ovom programu i za spremanje naziva funkcijskih tipki, te ga korisnik ne smije upotrebljavati ukoliko želi ispravan rad Editora)
80	Inicijalizacija programa
	PROCSWAP
120	Početne adrese spremnika za nazive funkcijskih tipki
130	Zamjena spremnika definicija
140-210	Zamjena spremnika naziva
220-230	Definiranje funkcijske tipke 0
240	Redefiniranje tipke TAB
250	Ako je MODEO poziva se PROCDISPLAY
	PROCDISPLAY
310-350	Prikazivanje naziva definicija funkcijskih tipki
370	Definiranje tekstovnog »prozora«
	PROCWINDOW
	Pomoćna procedura za crtanje »prozora« kod ispisivanja funkcijskih tipki
	PROCMENU
	Daje popis funkcija Editora i usmjerava program na izvršavanje izabrane funkcije
	PROCLIST
	Ispisivanje definicija funkcijskih tipki
	PROCCHANGE
	Redefinira odabranu funkcijsku tipku

750 Naredba OSCLI postoji u verziji BASIC2, a u starijim verzijama se može simulirati s:
$&CEO = "KEY" + KNUM
$ + KDEF$: x % = &EO: Y% = &C:CALL &FFF7
(vidi »USER GUIDE«, str. 463)

PROCPRTKEY

Ispisuje definiciju funkcijske tipke. Poziva FNLEN koja izračunava relativnu adresu zadnjeg znaka u »stringu« definicije funkcijske tipke

PROCSAVE

Sprema definicije i nazive funkcijskih tipki

PROCLOAD

Učitava definicije i nazive funkcijskih tipki

Napomena: naredba OSCLI u PROCSAVE i PROCLOAD može se simulirati na sličan način kao i u proceduri CHANGE

```
10*FX219,138
20FOR I%=&900TO&910:?I%=&10:?(I%+&100
)=&10:NEXT
30P%=&CC0:[OPT2:LDY#0
40.START:LDX &A00,Y:LDA &900,Y
50STA &A00,Y:LDA &B00,Y
60STA &900,Y:TXA:STA &B00,Y
70INY:BNE START:RTS:]
80PROCSWAP
90END
100:
110DEF PROCSWAP:LOCAL I%,L1%,L2%,L3%,T
1$,T2$,T3$
120L1%=&C00:L2%=&C40:L3%=&C80
130CALL &CC0
140FOR I%=1TO9
150T1$=$(L1%+I%*6):IF LENT1$>5 T1$=""
160T2$=$(L2%+I%*6):IF LENT2$>5 T2$=""
170T3$=$(L3%+I%*6):IF LENT3$>5 T3$=""
180$(L1%+I%*6)=T2$
190$(L2%+I%*6)=T3$
200$(L3%+I%*6)=T1$
210NEXT
220*KEY0 PROCSWAP|M
230$L1%="SWAP"
240*KEY10 MODE0:PROCMENU|M
250IF ?&355=0 PROCDISPLAY
260ENDPROC
270:
280DEF PROCDISPLAY:LOCAL V%
290V%=VPOS
300COLOUR0
310FOR I%=71TO-1 STEP-8
320PROCWINDOW(I%,3,I%+8,0)
330PRINTTAB(1,0);(I%+1)DIV8
340PRINTTAB(1)$(&C00+(I%+1)DIV8*6);
350NEXT
360COLOUR1:COLOUR128
370VDU28,0,31,79,4
380VDU31,0,V%
390ENDPROC
400:
410DEF PROCWINDOW(x,y,a,b)
420IF x=-1 x=0
430VDU28,x,y,a,b
440VDU17,128,12
450VDU28,x+1,y-1,a-1,b+1
460VDU17,129,12
470ENDPROC
480:
490DEF PROCMENU:LOCAL I%
500CLS:PROCDISPLAY
510PRINT'''TAB(7)"1) - PRIKAZIVANJE DE
FINICIJA FUNKCIJSKIH TIPKI"
520PRINT'TAB(7)"2) - DEFINIRANJE FUNKC
IJSKIH TIPKI"
530PRINT'TAB(7)"3) - SPREMANJE DEFINIC
IJA FUNKCIJSKIH TIPKI"
540PRINT'TAB(7)"4) - UCITAVANJE DEFINI
CIJA FUNKCIJSKIH TIPKI"
550REPEAT I%=VALGET$:UNTIL I%>0 AND I%
<5
560CLS
570IF I%=1 PROCLIST ELSE IF I%=2 PROCC
HANGE ELSE IF I%=3 PROCSAVE ELSE
PROCLOAD
580ENDPROC
590:
600DEF PROCLIST
610FOR I%=0TO9
620PROCPRTKEY(I%)
630NEXT
640ENDPROC
650:
660DEF PROCCHANGE:LOCAL KNUM$,KDEF$,LA
B$
670CLS
680PRINT'''TAB(7)"KOJA FUNKCIJSKA TIPK
A (1 DO 9, K=KRAJ) : ";:KNUM$=GET$:PRINT
KNUM$
690IF KNUM$="K" CLS:ENDPROC
700IF NOT(ASCKNUM$>48 AND ASCKNUM$<58)
GOTO670
710PRINT''TAB(7)"TRENUTNA DEFINICIJA :
"'
720PROCPRTKEY(VALKNUM$)
730PRINT''TAB(7)"UPIS NOVE DEFINICIJE
:"'
740PRINT"*KEY";KNUM$;:INPUT" "KDEF$
750OSCLI "KEY"+KNUM$+KDEF$
760PRINT'TAB(7)"NAZIV FUNKCIJSKE TIPKE
";KNUM$;:INPUT" ? (Do 5 slova!) : "LAB$
```

```
770IF LENLAB$>5 GOTO760
780$(&C00+VALKNUM$*6)=LAB$
790PROCDISPLAY
800GOTO670
810:
820DEF PROCPRTKEY(I%):LOCAL B%,F%,J%,N
%,A$
830PRINT"*KEY";I%;" ";
840B%=&B00+I%:A$="":@%=10:N%=?B%:F%=FN
LEN:FOR J%=N%+1TOF%:IF J%?&B00=&D A$=A$+
"!M" ELSE A$=A$+CHR$(J%?&B00)
850NEXT:IF F%=N% A$=""
860PRINTA$
870ENDPROC
880:
890DEF FNLEN:LOCAL H%,L%,R%
900H%=255:FOR L%=&B00TO&B0F:R%=?L%:IF
R%>N% AND R%<H% H%=R%
910NEXT:IF H%=255 H%=N%
920=H%
930:
940DEF PROCSAVE:LOCAL A$
950CLS:INPUT ''TAB(7)"NAZIV : "A$
960OSCLI "SAVE "+A$+" 900 D00"
970ENDPROC
980:
990DEF PROCLOAD:LOCAL A$
1000CLS:INPUT'''TAB(7)"NAZIV : "A$
1010OSCLI "LOAD "+A$+" 900"
1020CLS:PROCDISPLAY
1030ENDPROC
```

POKRETANJE PROGRAMA

Jednom kada je program upisan, treba ga renumerirati tako da početnoj programskoj liniji bude pridružen neki viši broj (npr. 30000), da bi korisnik mogao upisivati i svoje vlastite programe koji u tom slučaju moraju završavati naredbom END.
Program se pokreće komandom RUN. Pritisak na tipku TAB će inicijalizirati Editor na način kako je to prije opisano. U programu su korištene lokalne varijable, tako da korisnik ne mora voditi računa o nazivima varijabli u svojim vlastitim programima.
Treba napomenuti da je spremnik za defnicije funkcijskih tipki ograničen sa 256 bajtova. Prekoračenje njegove veličine uzrokuje ispis poruke »Bad key«. Zbog toga preporučamo korisnicima da definicije funkcijskih tipki upisuju u što sažetijoj formi.

Cvetković, dipl. inž.
S. Romac, dipl. inž.

COMMODORE-64, VIC-20,

RUTINA ZA ČEKANJE

Za vrijeme izvođenja često je potrebno zaustaviti program dok se ne stisne tipka. Kao primjer navodimo čitanje uputa na ekranu. Nakon što se prvi put stisne tipka, na ekranu se pojavi dio upute (koliko odjednom stane na ekran). Dok čitamo tekst, program »čeka«, a kad ponovno pritisnemo tipku, na ekranu se ispiše nastavak upute.

Navodimo dva načina kojima aktiviramo čekanje i prekidamo ga pritiskom na tipku.

```
10?«Stisnite bilo koju tipku«
20 GETA$
30 IF A$ = ""THEN 20
```

Druga rutina je isto tako poznata:

```
10? »Stisnite bilo koju tipku«
20 POKE 198,0
30 WAIT 198,1
40 PQKE 198,0
```

Kod upisivanja rutina možete izostaviti praznine i povezati naredbe u rečenice radi bržeg izvođenja.

RUTINA ZA PROŠIRENJE

Ponekad hoćete program za standardni 5K VIC izvesti na proširenom VIC-u od 8 ili 16K RAM-a.
Prelistajte sve programe i izbrišite POKE naredbe 52, 53, 55 ili 56. Nakon toga spremite program sa SAVE naredbom na kazetu.



COMMODORE

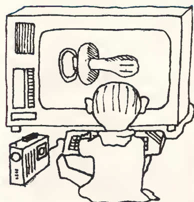
Prije unošenja nove verzije programa s kazete u memoriju treba ukucati:

```
10 POKE 642,32:
REM PROMJENA POCETAK BASICA
20 POKE 36869, 240:POKE 36866,150:
REM REORGANIZACIJA EKRANA
30 POKE 0,108: POKE 1,0: POKE2,192:
SYS0: REM RESTART BASICA
```

PREKLAPANJE PROGRAMA

Ponekad treba neki program puniti u memoriju pod kontrolom programa za kopiranje. Primjer: treba odabrati jedan iz niza programa. Upotreba naredbe LOAD za ovakvu vrsta programa najčešće je neupotrebljiva.

```
10?"[CLR]    LOAD"+CHR$(34)+"FILENA
ME"+CHR$(34)+",1"
20?"[CSR DWN] [CSR DWN] [CSR DWN]
RUN"
30 POKE198,3:POKE632,19:POKE
633,13:POKE634,13
```



GRAFIKA VISOKE RAŠČLANJENOSTI (RESOLUTION) NA STANDARDNOM VIC-u

Ovaj program pokazuje da se i na standardnom računalu bez posebnih dodavanja u RAM memoriju može koristiti grafika visoke raščlanjenosti.

Program crta krug i pravac.

```
10 REM SET HIRES SCREEN
15 POKE 36869,240 OR 13
20 POKE 36864,18
30 POKE 36865,46
40 POKE 36866,16 OR 128
50 POKE 36867,32
60 POKE 56,22: CLR
70 FOR X = 0 TO 255:
   POKE 7680+X,X:
   POKE 38400+X,0: NEXT
80 FOR X = 5120 TO
   5120+256*8: POKE X,0:
   NEXT
190 RD = 25: GOSUB 600
195 Y1=0: Y2=16*8-1:
    X1=0: X2=16*8-1:
    GOSUB 700: END
500 X% =X/8: Y% =Y% /8:
    P=X% +Y% *16+7680:
    CP=PEEK(P)
510 CH = 5120+CP*8+(Y
    AND 7): POKE CH,
    PEEK(CH) OR (2 (7-(X
    AND 7))): RETURN
600 REM DRAW CIRCLE
610 FOR Z= 0 TO 2* STEP.03:
    X = COS (Z)*RD+64: Y
    = SIN (Z)*RD*1.7+64:
    GOSUB 50 0:NEXT:
    RETURN
700 M = (Y2-Y1)/(X2-X1):
    FOR X=X1 TO X2: Y
    =M*X+C: GOSUB500:
    NEXT: RETURN
```

NAŠA PREPORUKA!

MATHS INVADERS Stell Saftware CBM 64

Sva se djeca vole igrati, ali samo neka vole i učiti. Zaključak se nameće sam: djeca najlakše uče igrajući se. Autor ove igre očito je istog mišljenja pa je izgradio igru pomoću koje će djeca vježbati matematiku.

Prema standardnom scenariju potrebno je zemlju spasiti od najezde neprijateljskih brodova iz svemira. Brodovi se obaraju točnim odgovorom na zadani matematički problem. Ako smo točno pogodili čuje se prasak pucnja i brod eksplodira. Brzina odgovora je važna, jer u slučaju sporosti pojavljuje se nova eskadrila. U jednom naletu dolijeće 15 ratnih brodova i kad ih sve uništimo sirena oglašava uspješno obavljanje zadatka. Postoje četiri vrste matematičkih radnji, na tri težinske razine: zbrajanje, oduzimanje, množenje i dijeljenje. Igra je tehnički izvedeńa dobro kako grafički tako i zvukovno. Premda se na tržištu mogu pronaći i atraktivnije akcione igre one vas nažalost ničem ne poučavaju. Zbog toga naš glas dajemo čvrsto za **MATHS INVADERS.**

commodore

64K

```
5 REM HI-RES ON THE 64
10 VC=13*4096+13*256:V=13*4096:POKE53280,0:POKE53281,1
15 PRINT"■■■■■SORRY TO KEEP YOU WAITING I WILL BE AS ■
   QUICK AS IS POSSIBLE."
16 POKE53280,6
20 FORX=0TO8192:POKEX+24*1024,0:NEXT
25 PRINT"■■■■■I WON'T KEEP YOU WAITING MUCH LONGER.":POKE53280,2
30 FORX=0TO1024:POKEX+18*1024,11:NEXT
40 POKEVC,PEEK(VC)AND254:REM SET UP VIC CHIP
50 POKEV+24,8:REM CHANGE VIDEO MATRIX BASE
80 POKEV+17,PEEK(V+17)OR32:REM SELECT BIT MAP MODE
70 W1=5:W2=50:POKE53280,0
75 FORG=1TO30
80 FORS=0TO6.5STEP.02:X=W1*SIN(S)+180:Y=W2*COS(S)+100
90 X=INT(X+.5):Y=INT(Y+.5):GOSUB2000
100 NEXTS:W1=W1+5:W2=W2-5:NEXTG:POKE53290,11:END
1999 REM*****PLOT SUBROUTINE**********
2000 Y1=INT(Y/8):Y2=Y-Y1*8
2010 X1=INT(X/8):X2=X-X1*8
2020 CH=(Y1*320)+(X1*8)+Y2:B1=2↑(7-X2)
2030 POKE24*1024+CH,PEEK(24*1024+CH)ORB1:RETURN

READY.
```

Program pokazuje mogućnosti grafike visoke raščlanjenosti. Isprobajte ga, neće vam biti žao.

64K

```
0 REM ***TO READ JOYSTICK PORTS ON THE 64
10 INPUT"■■■ WHICH JOYSTICK ONE OR TWO.";J
20 IF J>2 OR J<1 THEN 10
30 IF J=2 THEN 90
40 PRINT CHR$(5);"■■ THIS IS READING JOYSTICK ONE."
50 PRINT"■■■■■■■■"
60 S=NOT PEEK(56321)AND15:F=PEEK(56321)AND16
70 GOSUB 140
80 GOTO 60
90 PRINT CHR$(5);"■■ THIS IS READING JOYSTICK TWO."
100 PRINT"■■■■■■■■"
110 S=NOT PEEK(56320)AND15:F=PEEK(56320)AND16
120 GOSUB 140
130 GOTO 110
140 IF (S AND 1) THEN PRINT TAB(17);"UP    □"
150 IF (S AND 2) THEN PRINT TAB(17);"DOWN  □"
160 IF (S AND 4) THEN PRINT TAB(17);"LEFT  □"
170 IF (S AND 8) THEN PRINT TAB(17);"RIGHT □"
180 IF F=0 THEN PRINT TAB(17);"FIRE  □"
190 IF S=0 THEN PRINT TAB(17);"      □"
200 RETURN
READY.
```

Program za čitanje ulaznih informacija sa palica (joystick). Upotrebljavamo za programiranje igara za palice.

Pripremio: Željko Pađen

Program za brzo učitavanje i zapis na kazetu ZX 81 (16K)

Tko se od vlasnika kućnog računala ZX 81 nije barem jednom naljutio na svog miljenika satima i satima uzaludno nastojeći učitati neki program sa kazete?

Učitavanje za duže programe može potrajati i petnaestak minuta, a tada, malo oštećenje na traci ili neoprezan pokret rukom po tastaturi uništavaju sav vaš posao. Oni s boljim živcima pokušat će ponovo, dok će drugi samo zlovoljno odbaciti sve nade i prvom prilikom nabaviti bolje računalo. Za one koji su ostali vjerni starom dobrom ZX 81 objavljujemo program koji bitno skraćuje vrijeme potrebno za spremanje programa na kazetu (*SAVE*) i njegovo pravo učitavanje (*LOAD*).

Program posjeduje i mogućnost provjere upravo spremljenog programa na kazetu (*VERIFY*). U brzinu programa i njegovu efikasnost najbolje će se uvjeriti oni koji ga upišu u svoj ZX 81 i započnu izvoditi. Treba napomenuti da je zbog brzine snimanja potreban kvalitetniji kazetofon. Program ima ugrađenu rutinu za automatsko postavljanje vrha memorije (*RAMTOP*). Kada je program ukucan, dovoljno ga je učitati sa kazete, na početku rada s računalom. Program automatski postavi RAMTOP, preseli se sam u zaštićeno područje (iznad *RAMTOP-a*), te je spreman za upotrebu. Dio programa u BASIC-u koji je bio učitan sa kazete može sè tada prebrisati čak i naredbom NEW. Program će ostati u memoriji sve do gašenja napajanja za ZX 81. Dužina programa je oko 1/2 K, za koliko se efektivno smanji količina memorije za korisničke programe.



Da bismo program, koji je kompletno napisan u strojnom kodu, unijeli u računalo, potrebno je načiniti naredbu REM sa 50 znakova. Editiranjem treba načiniti desèt takovih REM naredbi.

(Slika br. 1).

```
   1 REM 12345678901234567890123
4567890123456789012345567890
   2 REM 12345678901234567890123
4567890123456789012345567890
   3 REM 12345678901234567890123
4567890123456789012345567890
   4 REM 12345678901234567890123
4567890123456789012345567890
   5 REM 12345678901234567890123
4567890123456789012345567890
   6 REM 12345678901234567890123
4567890123456789012345567890
   7 REM 12345678901234567890123
4567890123456789012345567890
   8 REM 12345678901234567890123
4567890123456789012345567890
   9 REM 12345678901234567890123
4567890123456789012345567890
  10 REM 12345678901234567890123
4567890123456789012345567890
```

Vrlo je važno da svaka od njih ima 50 znakova. Kada smo to učinili, treba pakirati slijedeće lokacije:

```
POKE 16512,2
POKE 16511,44
POKE 16510,0
```

Slika 2

Nakon toga treba upisati BASIC dio programa (slika 3). od linije 9000 do 9095.

```
9000 PRINT "STARTNA ADRESA "
9010 INPUT X
9015 CLS
9020 LET A$=""
9025 PRINT X;":";
9030 FOR A=1 TO 8
9035 IF A$="" THEN INPUT A$
9040 IF A$="S" THEN STOP
9045 PRINT A$( TO 2);" ";
9050 POKE X,16*CODE A$+CODE A$(2
)-476
9060 LET A$=A$(3 TO )
9070 LET X=X+1
9080 NEXT A
9090 PRINT
9095 GOTO 9025
```

Slika 3

Ovaj program služi za unošenje strojnog koda (slika 4.).

```
16514:76 FF 21 1C 7E 22 04 40
16522:2B 36 3E 2B F9 28 28 22
16530:02 40 21 A2 40 11 1C 7E
16538:01 E4 01 ED 80 C3 76 06
16546:CD A6 7F DD 26 00 CB 12
16554:C6 0A CD 28 7E 18 F8 0E
16562:01 42 3E 7F DB FE D3 FF
16570:1F 30 25 17 17 38 63 10
16578:F1 F1 AF 82 20 24 DD 84
16586:CA C3 03 E5 CD D2 7F 30
16594:37 2C 38 26 76 00 26 29
16602:37 9B E1 CD F4 7F CF FF
16610:AF DD B4 20 F9 82 28 E0
16618:CF 0C CD 28 7E 79 FE FF
16626:20 B8 CD 28 7E 79 FE 44
16634:20 B0 62 68 CD 28 7E C8
16642:7A 79 20 03 BE 20 A3 23
16650:17 30 F1 21 09 40 16 00
16658:CD 28 7E AF DD B4 79 20
16666:01 77 8E 20 AE CD F9 7E
16674:18 EC D8 FE 17 38 FB 06
16682:00 04 CA 2B 7E D8 FE 17
16690:50 F7 3E 26 B6 DA 2B 7E
16698:3E 0F B8 CB 11 D2 2D 7E
16706:C9 CD A6 7F DA F4 02 D5
16714:D5 CD E1 7F CD AB 7F E1
16722:16 0C 3E 7F DB FE 1F D2
16730:5A 7E 10 FE 1B 7A B3 20
16738:F1 1D CD 3E 7F 1E 44 CD
16746:3E 7F CD 3D 7F CB 7E 23
16754:26 F6 21 09 40 CD 3D 7F
16762:CD F9 7E 18 F8 23 EB 2A
16770:14 40 37 ED 52 EB D0 F1
16778:CD 28 7F AF DD 84 20 46
16786:CD A8 7F A7 ED 42 20 02
16794:CF FF CD BB 02 24 D3 FF
16802:28 F8 3E 7F D8 FE 1F D2
16810:C3 03 CF 14 3E 7F D8 FE
16818:1F 30 F7 06 19 10 FE CB
16826:13 9F E6 28 C6 1E D3 FF
16834:C9 5E 16 07 CD 28 7F 47
16842:10 FE 15 20 F7 CD 28 7F
16850:C6 09 47 10 FE C9 17 38
16858:3C E1 E5 CD D7 7F 3E 76
16866:D7 3E 76 D7 01 09 40 2A
16874:14 40 A7 ED 42 CD F4 7F
16882:CD D2 7F 00 27 37 34 2F
16890:00 27 3E 39 34 3B 26 76
16898:76 34 3B 2F 2A 37 26 8F
16906:CD 29 02 CD E1 7F DD 26
16914:FF D1 C3 26 7E CD D2 7F
16922:33 2A 32 26 00 2C 37 2A
16930:38 26 30 26 00 00 80 C3
16938:5A 7E E7 E7 CD A8 03 D5
16946:F5 2A 78 40 E5 01 00 00
16954:ED 43 78 40 11 09 40 1A
16962:81 4F 1A A8 47 13 37 2A
16970:14 40 ED 52 30 F1 ED 43
16978:78 40 E1 F1 D1 C9 E1 CD
16986:D7 7F E9 7E E5 7F D7 CB
16994:7E 23 28 F7 C9 FD CB 3B
17002:86 21 FF FF 22 34 40 22
17010:25 40 22 27 40 C3 2A 0A
17018:D5 E5 01 F0 D8 CD E1 07
17026:C3 AB 0A 00 00 00 00 00
```

Slika 4

Izvođenjem programa sa RUN 9000, započinje unos strojnog koda od početne adrese 16514 do 17037. Kad se dostigne adresa 17034, treba pritisnuti tipku S i time se zaustavlja unos koda.

U slučaju da smo u toku unosa programa pogriješili, zaustavljamo ga pritiskom na tipku S, a sa RUN se pokreće od one adrese koja je zadnja bila ispravno unesena. Jednom utipkani kod dobro je nekoliko puta spremiti na kazetu. Nakon toga utipka se BASIC dio programa (slika 5), i on je spreman za upotrebu.

```
0 REM
10 SAVE "PROG"
20 RAND USR 16516
```

Slika 5

Pritisnemo RUN i aktiviramo kazetofon za snimanje kako bismo program spremili na kazetu. Program se sam aktivira, učitava se sa LOAD"" i odmah nakon toga je spreman za upotrebu.

Nakon što je program učitan, obrišemo memoriju sa:

NEW (N/L).

Sada možemo pisati svoje vlastite programe ili učitati postojeće sa kazete. U trenutku kada želimo svoj program spremiti na kazetu u brzom načinu učinimo slijedeće:

RAND USR 32445 + CODE "ime programa"

zamjenjuje naredbu SAVE "ime programa" dok

RAND USR 32284 + CODE "ime programa"

zamjenjuje naredbu LOAD "ime programa".

Ukoliko smo zaboravili ime programa ili ga ne znamo, vrijedi također

RAND USR 32284 + CODE""

Nakon što smo ukucali i ostavili kazetofon spreman za snimanje

RAND USR 32445 + CODE" ime programa"

pritisnemo N/L (NEW/LINE). Nakon nekih 5 sekundi tišine (siva slika na ekranu), pojave se uske paralelne linije što znači da se program snima na kazetu.

Po završetku snimanja na ekranu se pojavi

ime programa
broj BROJ BYTOVA
OVJERA?

Pomaknemo li se na početak trake gdje smo počeli snimati program i pritisnemo li bilo koju tipku osim BREAK, izvršit ćemo provjeru upravo zapisanog sadržaja na kazetu. U slučaju ispravnog zapisa pojavit će se poruka:

NEMA GREŠAKA

U slučaju neispravnog zapisa:

KRIVA ADR. i adresa na kojoj je došlo do greške.

U ovakvom slučaju potrebno je ponovo izvršiti upis na kazetu na opisani način.

Damir Šegović

Isprobajte! ZX-81



Osnovna poteškoća koja se javlja kod programiranja računala jest da ono nikada ne radi ono što biste htjeli, već ono što ste mu programiranjem kazali da učini. Logičke greške, naročito kod većih programa, su neminovne, pa kod izvođenja rezultiraju onim što se u kompjutorskom žargonu naziva GIGO, odnosno smeće unutra – smeće van (Garbage in Garbage Out). Međutim, ponekad se i pažljivom analizom programskih nizova ne može ustanoviti zašto je rezultat izvođenja programa baš takav kakav jest, a ne onaj koji bi po svim pravilima programiranja trebao biti. Uzroke takvog ponašanja programa i računala treba tražiti u njegovom upravljačkom sustavu, odnosno u tumačenju sadržaja ROM-a takvih neuobičajeno postavljenih i sastavljenih naredbi. Pronalaženje situacija u kojima se računalo ponaša drugačije od očekivanog, često je rezultat slučaja ili upornog i dugotrajnog »hakerskog« ispitivanja računala. Naravno, ne treba uvijek kad dobijete GIGO rezultat pomisliti da ste naišli baš na takav slučaj. Treba pažljivo ispitati, korak po korak, izvođenje programa, pa ako ste stvarno pronašli nešto »novo«, pošaljite nam.

Rado ćemo to objaviti.

U današnjoj rubrici »Isprobajte« pokazat ćemo nekoliko primjera za vlasnike ZX81 računala u kojima će se njihov ljubimac ponašati drugačije od očekivanog. Za početak skinite svoja memorijska proširenja (ali prije toga isključite računalo iz napajanja!), tako da računalo posjeduje samo 1K RAM-a, stavite ZX81 u SLOW način rada i upišite

10 REMZX81

Nakon toga, kao direktne komande, unesite

POKE 16513,63
POKE 16517,147

Još jednom pritisnite NEW LINE i pokušajte objasniti dobiveni rezultat. Sličan efekt mogu dobiti i vlasnici starih ZX80 računala ukoliko upišu

10 REM ZX80

(Nastavak na 56 str.)

Igra koja dolazi

KOKOTONI WILF
»ELITE« – Spectrum 48K

Legenda kaže da je prije nekoliko stotina godina u sjevernoj Evropi živio veliki čarobnjak po imenu Ulrich.

Jednom je, proučavajući stare zapise slučajno otkrio da je nekad davno postojala zmajevska zaštitnička amajlija. Međutim, dijelovi te amajlije bili su razbacani kroz vrijeme, pa ih je trebalo sve sakupiti. Ulrich je bio prestar za tu avanturu te na put pošalje Kokotoni Wilfa, opremivši ga i parom krila.

Tu počinje i avantura, pomalo nalik na Manic Minera. Trebate proći nekoliko vremenskih perioda i pri tome sakupiti sve dijelove zaštitničke amajlije. Na putu vam, kao i obično, prijete mnoge opasnosti.

Avanturu počinjete 1.000.000 godina prije naše ere. Prolazite kroz čudnovat krajolik nastanjen dinosaurusima, pterodaktilima i drugim čudovištima. Trebate sakupiti dijelove amajlije i paziti da vas koje čudovište ne zahvati. Kad sakupite sve dijelove amajlije koji se nalaze u tom vremenu, potrebno je da se vratite istim putem i pokupite još jedan dodatni dio koji će vas prebaciti u slijedeći vremenski period. Sistem je isti u svim periodima. U godini 1066. prolazit ćete kroz srednjevjekovni dvorac i njegov podrum, i pri tom treba paziti na strelice, vojnike i druge opasnosti.

Tek kad stignete u budućnost (uskoro će biti prošlost) doznat ćete zašto Ulrich želi sastaviti amajliju. Odlična grafika i animacija likova učinit će vam zanimljivije putovanje kroz više razina igre. Da biste stigli do cilja potrebno je dosta spretnosti.

Želimo vam sretan put i ugodnu šetnju kroz vrijeme!

N.P.

avili i nagraditi.
lje ograničen, bitno je
najširem krugu koris-

50,5: DRAW

acija trokuta

50,5: DRAW

acija zvijezde

OT 150,5:

acija kvadrata

```
b = 22928:
12: FOR e = 0
T f = INT (1
: POKE b–e–
32*e,f: POKE
-33*d + 31 + 3
2*e,f: POK
```

Primjer 5. Zvučni uči

```
1 DATA 17,4,0,6,222,
29,197,213,205,181,3
35,16,244,201: FOR r
0021: READ m: POK
FOR x = 1 TO 30: RA
01: NEXT x
```

Primjer 6. Zvučni učir

```
1 DATA 6,2,197,17,4
1,205,181,3,17,48,0,3
5,181,3,193,16,234,2(
001 TO 30025: READ
NEXT n: FOR x = 1 T
USR 30001: NEXT x
```

Primjer 7. Zvučni uči

Keli



(Nastavak sa str. 54)

a zatim POKE 16431,587 i pritisnu NEW LINE.

Isključite, pa ponovno uključite računalo, jer tipka BREAK ne prekida prethodni primjer, pa upišite slijedeći program:

```
10 PRINT.;
20 GOTO 10
```

Pokušajte pretpostaviti rezultat. Utipkajte RUN i izvedite program.
Da li se vaše predviđanje ispunilo?
Unesite sada treći primjer i upišite RUN

```
1 FAST
2 SLOW
3 FAST
4 POKE 16427,1
5 CONT
```

Tipkom BREAK prekinite izvođenje programa, upišite NEW i unesite:

```
10 POKE 16390, INT(RND*256)
20 INPUT A$
30 PRINT A$
40 GOTO 10
```

Izvedite program i na upit računala unesite bilo koji znak ili znakovni niz, pa pritisnite NEW LINE.
Slijede još dva primjera za ZX80.

```
5 PRINT 1
10 POKE 16427,1
15 DIM A (:)
20 LET B = 128
25 INPUT C$
30 GOTO 5
```

Unesite bilo koji znak il
LINE.
Ponavljajte postupak d
program.
Slijedeći primjer daje s

```
10 PRINT 1;
20 POKE 16427,14
30 RUN
```

Utipkajte RUN i pritisnite NEW LINE. Kad se na ekranu pojavi poruka o grešci, pritisnite bilo koju tipku osim SHIFT, zatim RUN i NEW LINE.

Vratimo se ponovno ZX81 računalu. Unesite

```
10 GOSUB 10
20 PRINT »A«;
30 RETURN
```

Pokrenite program i pričekajte dok se ne pojavi poruka o grešci.
Tada, kao direktnu naredbu, upišite RETURN i pritisnite NEW LINE.
Kad se ekran popuni, pokušajte ponoviti postupak.

Sada, uz isključeno računalo, priključite memorijska proširenja i utipkajte:

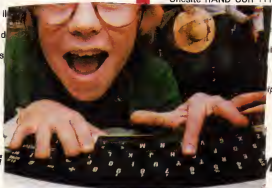
```
10 GOSUB 10
20 PRINT "A";
30 PAUSE B
40 RETURN
100 FAST
110 POKE 16389, 225
120 CLS
130 LET B = 1
140 GOTO 10
```

Kad unesete program, upišite RUN 100 i izvedite ga. Kad ugledate poruku o grešci, utipkajte RETURN i NEW LINE pa promatrajte kojom se brzinom popunjava ekran.

Za kraj vam poklanjamo jedan zanimljivi efekt. Unesite RAND USR 11111 kao direktnu koman-

za rubriku »Isprobajte«
programe kojima se po-
imljivi efekti za bilo koji

ipremio: Tomislav Žganec



Programi

Programiranje je umij
na relativno davno, uja se javija kod programi-
rad na računalima bino nikada ne radi ono što
broju ljudi, danas dobo ste mu programiranjem
nju. Tome je nesum greške, naročito kod ve-
razvoj mikroračunala inovne, pa kod izvođenja
mokratizacija progran kompjutorskom žargonu
ražavanja, postala je smeće unutra – smeće
današnjice. ge Out). Međutim, pone-
lizom programskih nizova
Razmotrimo li stvarnošto je rezultat izvođenja
programa kao izuzetnkav jest, a ne onaj koji bi
reg kruga korisnika ramiranja trebao biti. Uz-
uvjetima, u obzir dolaprograma i računala treba
grami, Eventualni nevljačkom sustavu, odnos-
pri tom tako presuda ROM-a takvih neuobiča-
govarajućih priključniavljenih naredbi. Pronala-
sketa) koji su neophodni za razvoj svakog ambicioznijeg projekta. Ograničavanje područja izražavanja na samo male programe ne mora umanjiti njihovu vrijednost. Elegan-

roprogramska rješenja
slučaj. Treba pažljivo isto se s uspjehom pri-
vođenje programa, pani dijelovi kompleksnih
nešto »novo«, pošaljitememorijski prostor, ili
Rado ćemo to objaviti.edbi predstavljaju redo-
U današnjoj rubrici »ištom autoru, pri čemu
nekoliko primjera za zi do punog izražaja. S
kojima će se njihov lprogrami su, s obzirom
od očekivanog. Za poristupačni najširem kru-
rijska proširenja (ali pstveno se misli na po-
iz napajanja!), tako dam i testiranjem pojedi-
1K RAM-a, stavite Zkih rješenja mogu us-
upišite
10 REMZX81 saznanja.
Nakon toga, kao direko ovaj oblik stvaralaštva
POKE 16513,63 korisnika mikroračunala,
navodimo nekoliko primjera mikroprograma pisanih u jednoj jedinoj programskoj liniji (koja se naravno može sastojati iz više naredbi, shodno pravilima BASIC-a). Programi su pisani za računalo Sinclair ZX Spectrum i



mogu poslužiti kao svojevrsni prikaz rada pojedinih dijelova računala (grafika, zvuk), ili kao dijelovi većih programa.

Pozivamo sve zaintersirane korisnike da nam pošalju svoja originalna rješenja (pisana kao program u jednoj liniji). Najinteresantnije priloge ćemo objaviti i nagraditi.

Izbor problematike nije ograničen, bitno je da bude interesantna najširem krugu korisnika.

```
1 LET x = 524: PLOT 150,5: DRAW
0,150,x
```

Primjer 1. Grafika – rotacija trokuta

```
1 LET x = 587: PLOT 150,5: DRAW
0,150,x
```

Primjer 2. Grafika – rotacija zvijezde

```
1 LET x = 1111115: PLOT 150,5:
DRAW 0,150,x
```

Primjer 3. Grafika – rotacija kvadrata

```
1 LET a = RND*222: LET b = 22928:
LET c= 0: FOR d = 1 TO 12: FOR e = 0
TO c: LET a = a + .03: LET f = INT (1
27*SIN a): POKE b + e,f: POKE b–e–
1,f: POKE b + 32–31*d + 32*e,f: POKE
b–31*d–32*e,f: POKE b–33*d + 31 + 3
2*e,f: POKE b–33*d–1–32*e,f: POK
E b64*d + 31–e,f: POKE b–64*d + 32 +
e,f: NEXT e: LET c = c+1: LET b= b +
32: NEXT d: GO TO 1
```

Primjer 4. Grafika – kaleidoskop

```
1 DATA 243,17,16,250,38,250,5
8,72,92,31,31,31,14,254,238,16,2
37,121,67,16,254,37,32,244,28,21,
32,232,251,201: FOR n = 30001 TO
30030: READ m: POKE n,m: NEXT n:
RANDOMIZE USR 30001
```

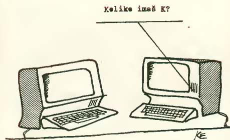
Primjer 5. Zvučni učinak – »star ship«

```
1 DATA 17,4,0,6,222,33,66,0,2
29,197,213,205,181,3,209,193,225,
35,16,244,201: FOR n = 30001 TO 3
0021: READ m: POKE n,m: NEXT n:
FOR x = 1 TO 30: RANDOMIZE USR 300
01: NEXT x
```

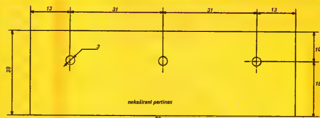
Primjer 6. Zvučni učinak – »laser gun«

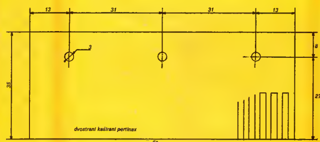
```
1 DATA 6,2,197,17,48,0,33,255,
1,205,181,3,17,48,0,33,144,2,20
5,181,3,193,16,234,201: FOR n = 30
001 TO 30025: READ m: POKE n,m:
NEXT n: FOR x = 1 TO 30: RANDOMIZE
USR 30001: NEXT x
```

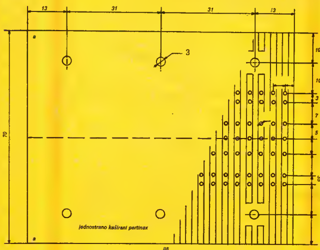
Primjer 7. Zvučni učinak – »flipper«



RUBNA SPOJNICA »Edge« connector za računala GALAKSIJA







SL. 4 MJERNE SKICE DIJELOVA SPOJNICE ZA GALAKSIJU

Veza mikroračunala s priključnim uređajima redovno se ostvaruje rubnim spojnicama (»edge« konektorima) koji obzirom na svoju jednostavnost i pouzdanost posve zadovoljavaju sve zahtjevane kriterije. Jedini nedostatak (koji se uostalom odnosi i na sav ostali sitni instalacioni materijal u vezi računala) je u relativno visokoj cijeni, koja čak može nadmašiti ukupnu cijenu svih ostalih dijelova (integrirani krugovi, otpornici itd.) uređaja koji želimo izgraditi, pa često odustajemo od projekta.

Rješenje koje nudimo namijenjeno je računalu Galaksija, prvom domaćem mikroračunalu za samograditelje. Predložena izvedba može se prilagoditi i za druge standarde, odnosno druga mikroračunala. Ostvareni rezultati posve će zadovoljiti amaterske zahtjeve, dok je cijena upotrijebljenog materijala gotovo zanemariva.

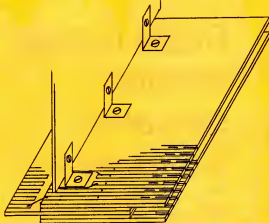
Osnovni materijal za izvedbu (rubne spojnice) je kaširani »pertinaks« (jednostrani i dvostrani) debljine 1,50 mm, te mesingana ili bakrena pokositrena žica promjera 0,8 do 1,0 mm. Obzirom na relativno male dimenzije spojnice, posve će zadovoljiti otpaci navedenih materijala.

Praktična izvedba spojnice prikazana je slikama. Čitav sklop je izveden kao propusna rubna spojnica, što znači da se nekoliko ovakvih spojnica s priključenim dodatnim uređajima mogu nadovezivati jedna na drugu. Dodatni uređaj koji njome povezujemo može biti montiran na samu spojnicu ili povezan vrpcom vodiča.

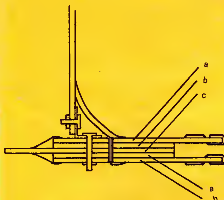
Na kraju dodajemo još nekoliko napomena za manje iskusne samograditelje.

Najprije se iz pertinaksa, prema mjerama na slikama, rezbarskom pilicom izrežu svi dijelovi (pločice) spojnice. Pri tom se gornja i donja pločica »ženskog« dijela spojnice (a) ostave u komadu radi lakšeg izvođenja štampanih veza.

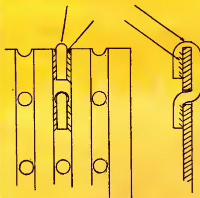
Pertinaks spreman za izvođenje štampanih veza pažljivo se očisti sa strane bakrene folije (npr. običnom mekanom gumicom za brisanje) i na foliju se precizno, prema nacrtu, ucrtaju olovkom položaji budućih vodiča te označe budući otvori (rupice).



Sl. 1 IZGLED GOTOVE SPOJNICE



Sl. 2 IZGLED SPOJNICE U PRESJEKU



Sl. 3 DETALJ IZVEDBE KONTAKTA

Na tako iscrtanu foliju nalijepe se zatim trake obične samoljepive vrpce (»selotejp«), jedna do druge, tako da cijela površina folije bude pokrivena. Ovaj posao treba izvesti pažljivo, treba obratiti posebnu pažnju na eventualne mjehuriće zraka ispod samoljepive vrpce. Po potrebi se na kritična mjesta može nalijepiti još pokoji sloj vrpce, dobra pokrivenost je izuzetno značajna za uspješno i uredno izvedenu štampanu vezu.

U slijedećoj fazi izrade, tapetarskim nožem (Olfa) ili običnim žiletom izreže se nalijepljena vrpca prema crtežu na foliji, te se pažljivo odlijepe dijelovi na onim mjestima koja trebaju biti nagrižena. Eventualne pogreške u ovoj fazi rada lako se ispravljaju ponovnim lijepljenjem samoljepive vrpce i pravilnim isijecanjem.

Pločica je spremna za nagrizanje izloženih dijelova bakrene folije. Jedan od najekonomičnijih recepata tekućine za nagrizanje je:

1/3 solne kiseline za čišćenje u domaćinstvu (28-32%)
1/3 vodikovog peroksida za upotrebu u domaćinstvu (30%)
1/3 obične vode.

Omjer pojedinih sastojaka nije kritičan. U posudu najprije ulijemo kiselinu, pa vodu, pa peroksid. Radi jake reakcije s bakrom poželjno je postupak izvoditi na otvorenom uz upotrebu predviđenih zaštitnih sredstava (gumene rukavice, hvataljke i sl.). Upotrijebljena posuda može biti staklena, proculanska ili plastična. Pripremljenu pločicu uronimo u smjesu za nagrizanje i nakon nekoliko desetaka sekundi ono je gotovo. Pažljivo izvadimo pločicu i ubacimo u posudu s običnom vodom kako bi se posve prekinulo djelovanje kiseline. Nakon naknadnog ispiranja u tekućoj vodi pločica je spremna za dalju obradu.

Nakon ispiranja i sušenja pločicu očistimo od ostataka samoljepive vrpce, izbušimo naznačene rupice i tek sada raspilimo (razdvojimo) gornju i donju pločicu »ženskog« dijela čime je ostvarena njihova potpuna simetričnost.

Štampana pločica »muškog« dijela konektora izvodi se na identičan način, pri čemu treba posebno paziti na poravnanost gornjih i donjih kontakata.

Preostalo je još lemljenje kontaktnih žica na gornji i donji dio »ženskog« dijela rubne spojnice, te sklapanje i ožičenje svih njenih dijelova u cjelinu, kako je prikazano na slikama.

Damir Flo

Fizičko brisanje memorije i sprečavanje pregrijavanja ZX SPECTRUMA

Problem potpunog čišćenja (resetiranja) memorije kod ZX Spectruma standardno se rješava izvlačenjem utikača za napajanje iz utičnice na tastaturi i ponovnim vraćanjem natrag. Ova metoda prije ili kasnije ima za posljedicu slabljenje kontakata utičnice i prekid u vodičima (da o gubitku koncentracije korisnika računala i ne govorimo). Rješenje je u ugrađivanju odgovarajućeg prekidača. Mnogi će se zapitati zašto nadahnuti konstruktori nisu ugradili i tu »sitnicu?« Odgovor je prozaičan: maksimum (programske) kvalitete uz minimalnu cijenu, što je kod Spectruma svakako postignuto. U Engleskoj se svaki, pa i najmanji, dodatak naplaćuje, pa su novine prepune reklama u kojima se nude najraznovrsniji dodaci.

Varijanta rješenja koje nudimo namijenjena je najširem krugu korisnika i ne zahtijeva nikakvo iskustvo iz praktične elektronike (nema lemljenja i otvaranja uređaja!). Pored toga ovo rješenje ujedno omogućuje otklanjanje još jednog problema: pregrijavanja.

Pregrijavanje Spectruma, koje u nekim slučajevima dovodi do privremenog gubitka funkcija procesora, uzrokovano je prvenstveno disipacijom razlike električne snage koja se privodi tastaturi iz mrežnog adaptera i stvarne snage koju samo računalo troši. Do spomenutog »viška« snage dolazi stoga što je napajanje većine priključnih uređaja (pisaća, međusklopa itd.) izvedeno preko samog računala – rubna spojnica (edge conector) na stražnjoj strani. Najjednostavnije rješenje problema je ugradnja odgovarajućeg predotpora u strujni krug između mrežnog adaptera i tastature.

SL 1 MONTAŽA »LUSTER« PREKIDAČA





U praktičnoj izvedbi koristit ćemo tzv. provodni prekidač (*»luster šalter«*), koji montiramo na vodič što spaja mrežni adapter i tastaturu nekoliko centimetara od utikača. Na ovaj način pri radu računala prekidač za potpuno čišćenje nam je u svakom trenutku na dohvat ruke i na neki način postaje sastavni dio računala (u njegovu neophodnost uvjerit ćete se i sami).

Ako želimo ugraditi i spomenuti predotpor za kompenzaciju pregrijavanja, nema za to pogodnijeg mjesta od unutrašnjosti prekidača. Predotpor mora imati vrijednost od cca 2 do 4 Ohm-a i podnositi trajnu struju od cca 0,50 A. Komadić »Cekas« žice (od spirale električnih grijača) promjera 0,30 mm i duljine cca 20 do 25 cm posve će odgovarati traženim zahtjevima. »Cekas« žicu namotamo u spiralu i spojimo u seriju s prekidačem (vodiče međusobno spojimo jednostavnim uplitanjem).

Svi detalji praktične izvedbe prekidača i predotpora prikazani su na slikama.

Ovo je tek jedno od velikog broja mogućih rješenja koje smo opisali zbog jednostavnosti izvedbe i materijala koji je lako nabaviti. Možda će netko radije u kutiju mrežnog adaptera ugraditi predotpor i mikroprekidač (*»kip-šalter«*).

Ukoliko imate rješenje koje smatrate efikasnim i jednostavnim, molimo, javite se.

D. F.

VIC-20

Kazetni adapter

Pripremio: Branislav Delibašić

Međusklop (interface) za mikroračunalo VIC-20, koji omogućuje upotrebu običnog kazetofona.

Obzirom da korištenje VIC-20 »digitalnog« kazetofona u sklopu mikroračunala predstavlja priličan izdatak, to upotreba ovog sklopa koji omogućuje korištenje običnog kazetofona za istu svrhu, može biti prihvatljivo rješenje.

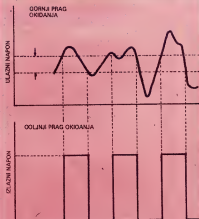
OPIS RADA SKLOPA

Sklop na slici 1 osniva se na upotrebi u elektronici poznatog »SCHMITTOVOG OKIDNOG SKLOPA«, ovdje u formi integriranog kruga IC1. Glavno svojstvo tog sklopa je da on pretvara nepravilni po valnom obliku ulazni signal, u slijed čistih pravokutnih impulsa u ovom slučaju TTL razine. Tipičan rad Schmittovog okidnog sklopa prikazan je na slici 2. Dioda D1 sprječava negativnu pobudu na ulazu sklopa. Kao što je prikazano na slici 2A Schmittov sklop ima gornji i donji prag okidanja. Kada ulazni signal prijeđe vrijednost gornjeg praga, izlazni signal postaje TTL visok i ostaje takav tako dugo dok ulazni signal ne padne ispod vrijednosti donjeg praga. Kada se to dogodi izlazni signal postaje TTL niske razine (OV). Na varijacije valnog oblika ulaznog signala u područje između pragova, sklop je neosjetljiv.

Sklop je moguće ostvariti na perforiranoj pločici ili na maloj štampanoj pločici. Integrirani krug IC1 sadrži u sebi šest Schmittovih sklopova: moguće je upotrijebiti bilo koje (dva od njih). Raspored nožica prikazan je na slici 3.

Napajanje i masa se dobivaju sa točke B i A respektivno sa priključka kazetnog kanala (casette port connector) mikroračunala VIC-20. Da bi se smanjio potrošak struje integriranog kruga potrebno je uzemljiti ostale slobodne ulazne nožice na IC1.

Kazetni je međusklop spojen na mikroračunalo preko kazetnog priključka (edge connector), koji može biti napravljen od bilo kojeg priključka sa razmakom 3.91mm (0.154 inča) između kontakata.



Sl 2. VALNI OBLIK ULAZNOG (A) I IZLAZNOG (B) SIGNALA U OVISNOSTI O VREMENU ZA SCHMITTOV OKIDNI SKLOP

POVEZIVANJE KAZETNOG MEĐUSKLOPA

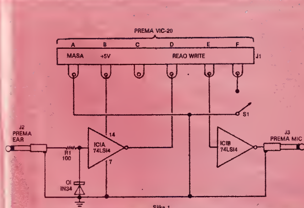
Priključak J1 kazetnog međusklopa potrebno je spojiti na priključak kazetnog kanala mikroračunala VIC-20. Izlaz kazetofona (ear) treba spojiti na priključak J2. J3 međusklopa treba spojiti na mikrofonski ulaz kazetofona.

Kod pohranjivanja programa na kazetu utipkajte »SAVE«. Računalo odgovara sa »Press record and play on the tape«. Pritisnite »Record« i »Play« taster na kazetofonu i aktivirajte sklopku S1 na međusklopu. Ako kazetofon ima regulaciju ulaznog signala potrebno je iznaći optimalan položaj.

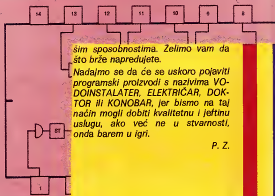
– Kod unošenja programa u mikroračunalo utipkajte »LOAD« i aktivirajte sklopku S1.

Napomena: Kada međusklop nije u upotrebi sklopka S1 treba biti otvorena.

Prema Computers & Electronics



Slika 1



šim sposobnostima. Želimo vam da što brže napredujete.

Nadajmo se da će se uskoro pojaviti programski proizvodi s nazivima VODOINSTALATER, ELEKTRIČAR, DOKTOR ili KONOBAR, jer bismo na taj način mogli dobiti kvalitetnu i jeftinu uslugu, ako već ne u stvarnosti, onda barem u igri.

P. Z.

TOP

RONILAC SCUBA

Ova igra, koja se u zadnje vrijeme probila u sam vrh najpopularnijih igara, djelo je manje poznate programske kuće Durrell Software (Harrier Attack). Odlikuje se kvalitetnom grafikom i vrlo dobrim crtežima likova.

Cilj je spustiti simpatičnog čovječuljka Scubu, čije zanimanje je ronilac, što dublje.

Čim se spusti u vodu, Scuba nailazi na razne opasnosti i prepreke: meduze, sipe, morske pse i ostale nemani koje mašu repom i razjapljuju čeljusti ne bi li se dočepale jadnog Scube.

Na dnu se nalaze školjke bisernice koje su cilj Scubinog ronjenja. Da zadatak ne bude previše lagan, biseri se mogu pokupiti samo dok je školjka otvorena. Školjku treba odnijeti na jahtu (koju pronađemo nakon što izronimo na površinu). Oslobodivši se dragocjenog tereta Scuba se ponovo vraća u morske dubine. Šećući po dnu Scuba nailazi na rupe u kojima su goleme hobotnice. Potrebna je određena vještina da se prođe kroz rupu i izbjegnu smrtonosni kraci hobotnice. Životinjski i biljni svijet naglo se mijenja, tako da su neke ribe i do pet puta veće od ronioca Scube. I školjke su mnogo veće. Ako se zaklope dok vadimo biser, Scuba pogiba i igra je završena. Želimo li se spustiti još dublje moramo proći rupu koju nastava hobotnica. Životinje postaju sve rjeđe, svega nekoliko meduza, ali ima mnoštvo rupa i špilja. Kad se spustimo do samog dna, uz malo sreće ugledat ćemo bocu s kisikom. Naslonimo li se na nju, za čas ćemo se snabdjeti za život nužnim kisikom. Pretražujući špilje pronalazimo zakopano blago koje je svrha naše avanture. Kako blago pod vodom ne vrijedi mnogo, potrebno je izroniti, pronaći jahtu i na njoj pohraniti blago. U tom trenutku nas rezultat upisuje se u SCORE i postaje punovažeći.

Na početku igre dobivamo određeni broj »životnih« poena. Igra traje dok ne potrošimo »sve svoje živote«.

KOMANDE:

Z – okretanje u smjeru kazaljke na satu

X – okretanje u smjeru suprotnom od kazaljke na satu

Symbol schift – kočenje

Break-space – ubrzavanje

Program ima dodatne mogućnosti:

K – KEY – promjena komandi

S – SAVE – automatsko snimanje programa

L – LOAD – učitavanje programa

Vanja Delibašić

Top 10

1.	SABRE WULF	Ultimate
2.	FLIGHT PATH 737	Anirog
3.	TRASHMAN	New Generation
4.	LORDS OF MIDNIGHT	Beyond
5.	BEACH HEAD	US Gold
6.	KOKOTONI WILF	Elite
7.	SCUBA DIVE	Durrell
8.	MUGSY	Melbourne House
9.	THE FALL OF ROME	APS
10.	JET SET WILLY	Software Projects



10

TRASHMAN – SMETLAR

Konačno se pojavio program koji možemo iskoristiti da bismo očistili naš bijeli Zagreb grad od smeća koje se godinama taloži i prijeti da prekrije pločnike.

Radi se o programu koji prevodimo sa čistač ili još sočnije sa smetlar. Dakle, smetlar započinje posao ispred kućnog praga programera Paula Buma koji živi u otmjenoj ulici Montaque Road za razliku od autora ovog članka koji stanuje u Sesvetama.

Pomoću palice za igru ili tipke kojom pomičemo pokazivač (cursor) vodimo našeg smetlara i pomažemo mu pri sakupljanju kanti s otpacima koje on prazni u smetlarski kamion što ga vozi ulicama voljenog grada.

Položivši kvalifikaciju jurimo ulicama radeći svoj posao. Posao nam ne bi bio tako težak kad ne bismo trebali paziti da ne zagazimo na posjede (travu) naših uglednih sugrađana koje čuvaju njihovi vjerni psi čuvari što jedva čekaju da zagrizu u naše meso. Ali naš najljući neprijatelj nisu životinje, već zločesti biciklisti i bezobzirni vozači automobila, koji (vjerojatno psujući nas) silom od nas pokušavaju napraviti bivšeg, tj. pokojnog, smetlara.

Koristeći svoje vozilo nastojimo se što duže održati u ovoj, životu toliko sličnoj, igri.

Shodno zakonima fizike, kad vozimo smeće krećemo se sporije, ali kad ispraznimo naš teret, možemo si dozvoliti vratolomije koje smo zapamtili gledajući naše dečke kako se utrkuju ulicama rodnoga nam grada.

Najviše nas ipak žalosti ili razjari (ovisno o temperamentu) činjenica da nam se smanjuje plaća (broj bodova) ukoliko stupimo nogom na posjed bolje stojećih građana. Radi pedagoške pouke kazna se pojačava jer u naše slobodno vrijeme ne smijemo ući u kuće i »u fušu« ponuditi naše usluge.

Svaka kuća raspolaže s četiri različite dosjetke, koje su njezini dokoni ukućani smišljali dok smo mi čistili ulice.

Razgovori su tipa: »Hoćete li pogledati moj televizor?«

Kad se lakomo pokušamo uvaliti u udobnu fotelju i predahnuti od jurnjave i posla, domaćin nas upozorava da smo ga krivo shvatili:

»Mislio sam da popravite televizor, a ne da gledate program«.

Igra ima sedam razina i u svakoj imamo po tri pokušaja (osim ako nas ne pregazi auto). Ubrzo razjašnjavamo tajanstvenost događaja; bicikli se pojavljuju u drugoj ulici, travnati ivičnjak u trećoj, a u petoj je birtija gdje nas čekaju raznovrsna iskušenja, od kojih je najveća opasnost od opijanja i prenajedanja koje nas jako usporava. U zadnje dvije ulice čak nas niti auti više ne ganjaju, ali nailazimo na pravi engleski »pub«. Na retoričko pitanje »Jedan putni gemišt«, (»One for the road?«), dobivamo odgovor:

»Vaš posao je da praznite kante, a ne bačve«, što dokazuje da Englezi bez obzira na tragičnost situacije ne gube smisao za humor. Ako uspijete odigrati do kraja, javite se Superhiku iz »Alan Forda« koji vam možda pronađe odgovarajući posao u New Yorku, jer ćete steći titulu diplomiranog smetlara: »Ovim se potvrđuje da ste uspješno završili tečaj za zvanje smetlara, te da možete početi čistiti svijet od smeća.«

Očigledno je da vaše napredovanje u službi ovisi jedino i isključivo o vašim sposobnostima. Želimo vam da što brže napredujete.

Nadajmo se da će se uskoro pojaviti programski proizvodi s nazivima VODOINSTALATER, ELEKTRIČAR, DOKTOR ili KONOBAR, jer bismo na taj način mogli dobiti kvalitetnu i jeftinu uslugu, ako već ne u stvarnosti, onda barem u igri.

P. Z.



NARUDŽBENICA **mr** br 1

PROSVJETA

Berislavićeva 10, P.P. 634,
41001 Zagreb
tel: 423-280 i 423-480

PREZIME I IME ....

ULICA I BROJ ....

BROJ POŠTE I MJESTO ....

NARUČUJEM KOD „PROSVJETE" UZ ODGOVARAJUĆE UVJETE PLAĆANJA

1. Jure Špiler: BASIC po cijeni od 950.– dinara s 10% popusta
2. Komplet „SVIJET KUĆNIH RAČUNALA" 1–4 u pretplati
   a) za plaćanje odjednom po cijeni od 3.200.– din.
   b) za plaćanje po cijeni od 4.000.– din. u 4 rate po 1.000.– dinara mjesečno

(Zaokružite broj varijante za koju se odlučite)

Datum: Potpis:

273
274 ABC
275

mr®

REVIJA ZA MALA RAČUNALA

Izdavač: NN RO »Sportska tribina« Zagreb, Radićeva 27

Za izdavaće: Josip Čondić

Izdavački savjet: Antun Krajnović, pradsjednik; Darko Fischer, mr; Mladan Glazenhardt, dipl. prav; Željko Kavran, dipl. ing; Branimir Mitrović; Vilko Žiljak, dr;

Glavni urednik: Željko Pažur, prof. i dipl. ek;

Redakcija: Željko Pažur, prof. i dipl. ek;
Damir Fio, dipl. inž;
Ernest Kogalman, dipl. inž;
Eduard Kovačić, dipl. inž;
Darko Paček, dipl. inž;
Tomislav Žganec, dipl. inž;

Naslovna strane: Ervin Šikić i Marijan Osman

Likovno i tehnićko uređenje: Marijan Osman

Crteži: Ninoslav Pozojavić i Ivica Čengić

Fotografija naslovne strane: Mario Mlaća

Sjedište redakcije: Zagreb, Radićeva 27/I

Telefoni: 274-987; 272-808

Žiro račun· 30102-603-10911

Revija izlazi dvomjesećno

Cijana pojedinom primjerku 250 din

Nenaručene rukopise i slika ne vraćamo.

Tisak: »Varteks« RO Tiskara – Varaždin

POZIV NA PRETPLATU

- **10% uštede**
- **garantirana cijena za cijelu godinu**
- **besplatna dostava na Vašu adresu**

Pretplatom na reviju **mr** ostvarujete gore navedene pogodnosti. Da biste postali pretplatnik potrebno je da izvršite uplatu potrebnog iznosa poštanskom uputnicom na našu adresu NN RO »SPORTSKA TRIBINA« (za **mr**) 41000 Zagreb, Radićeva 27/I ili bankovnom uplatnicom na naš žiro račun broj: 30102-603-10911 – »Sportska tribina«.

Potvrdu o izvršenoj uplati priložite uz narudžbenicu i pošaljite na našu adresu.

– cijena pretplate za godinu dana (6 brojeva) Din 1.350.–
– cijena pretplate za pola godine (3 broja) Din 675.–
– cijena pojedinog broja Din 225.–

Ime i prezime ....

Točna adresa ....

....

potpis ....



Prvi broj ste pročitali. Očekujemo Vaše primjedbe od kojih ovisi kako će izgledati drugi broj. Napišite sve što smatrate važnim bilo u dobrom ili lošem smislu. Molimo vas da nam javite svoje impresije, komentare, sugestije. Pošaljite nam članke ili priloge za koje smatrate da su zanimljivi za objavljivanje. Ukoliko ste autor programa ili neke interesantne rutine, javite se ili nam pošaljite materijale koje želite objaviti, uz odgovarajući honorar.

Adresa: NN RO »Sportska tribina« (za MR), 41000 Zagreb, Radićeva 27/I